Dynamic Forms 3.3

User Guide

Table of contents:

List of figures:

1 INTRODUCTION

1.1 What's new in v3.3?

- Now fully compatible with DNN 5.2.x!
- Quick Edits -- you can now change field settings and properties right on the user-facing page. No need to switch to Manage Question mode for most settings! Can also be disabled if preferred.
- New option to hide the module based on a querystring variable to support more complex work flow scenarios when integrating with other modules.
- SQL Binding now supports external connection string.
- New Question Event to automatically submit the form upon firing -- takes the place of the submit button when conditions are met!
- Text/HTML form fields can now be rendered on email and PDF events with a token.
- View Form Results now have expanded functionality!
    - Allow a non-admin role to view form results, manage results template, export to Excel and/or purge form results, just like an Admin.
    - Display results above or below the form --view form entries as they come in!
- New DNN Text Suggest field type allows you to build your own suggestion list to aid users as they fill out your form.
- Now apply SQL validation to individual questions, not just the entire form!

2 INSTALLATION PROCEDURE

Included in your download are either one (If you only purchased the PA) or two (If you purchased the PA and Source Code) zip files. One zip file is the source code to the application, and the other zip file is the module which you can upload to your site.

When you extract the files, you will notice that it extracts two zip files (**note:** only one if you just purchased the PA).

- **DataSprings_DynamicForms_Module30_DNN45PLUS.zip** - file for installing "Dynamic Forms" with your DNN
- **DataSprings_DynamicForms_Source.zip -** source zip file you can use in order to make any changes to the application (**note:** meant for advanced users)

In order to install your "Dynamic Forms" module, login with an account to your DNN® site as a host or administrator account. Once logged in, Navigate to the Host menu item, and click on the "Module Definition".

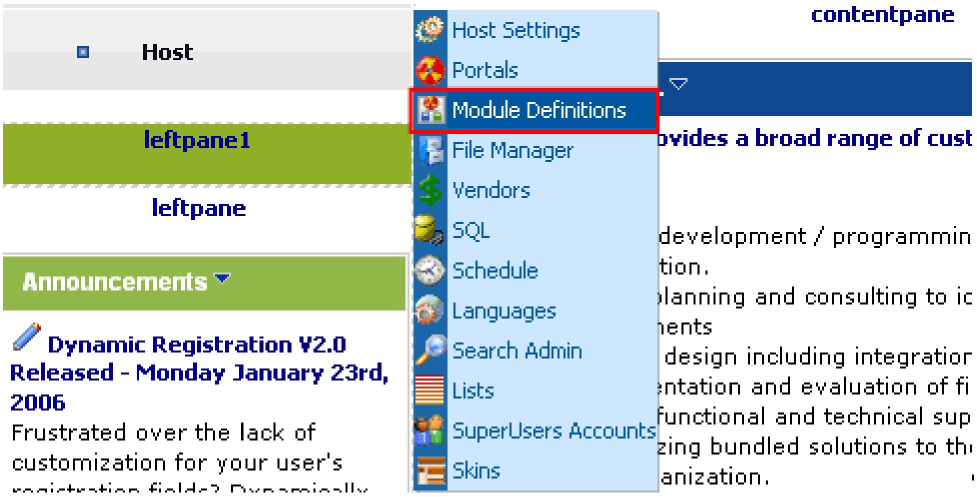


Figure 1: Installation procedure (step 1/6)

The following screen will be displayed.

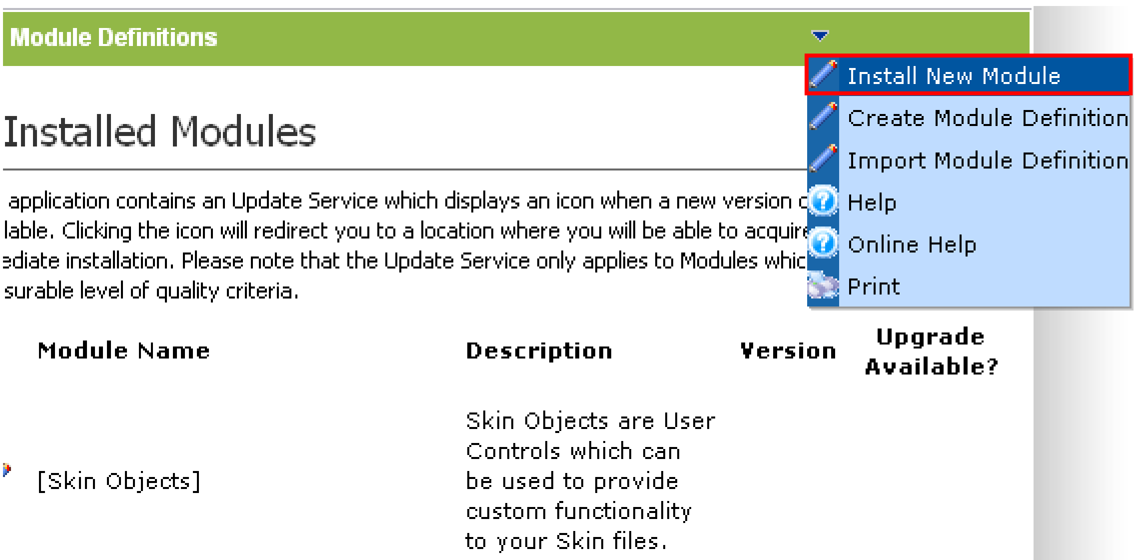


Figure 2: Installation procedure (step 2/6)

Click on the "Upload New Module" to continue installing "Dynamic Forms" and the following screen will be displayed.

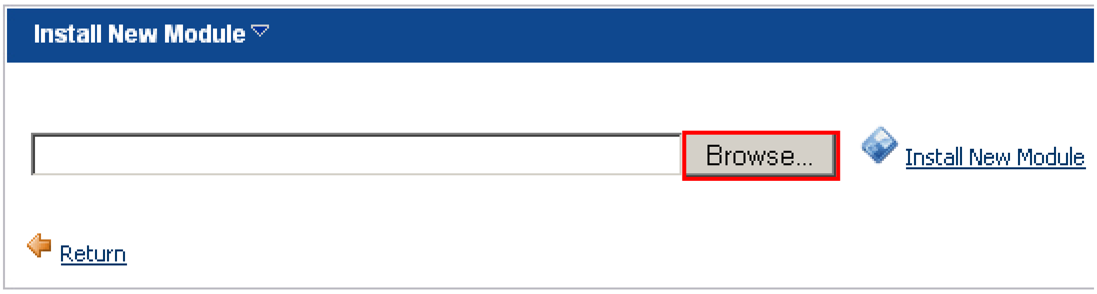


Figure 3: Installation procedure (step 3/6)

Click on the “Browse” button and the dialog window for locating the installation file “**DataSprings_DynamicForms_Module27_DNN45PLUS.zip**” will be displayed.

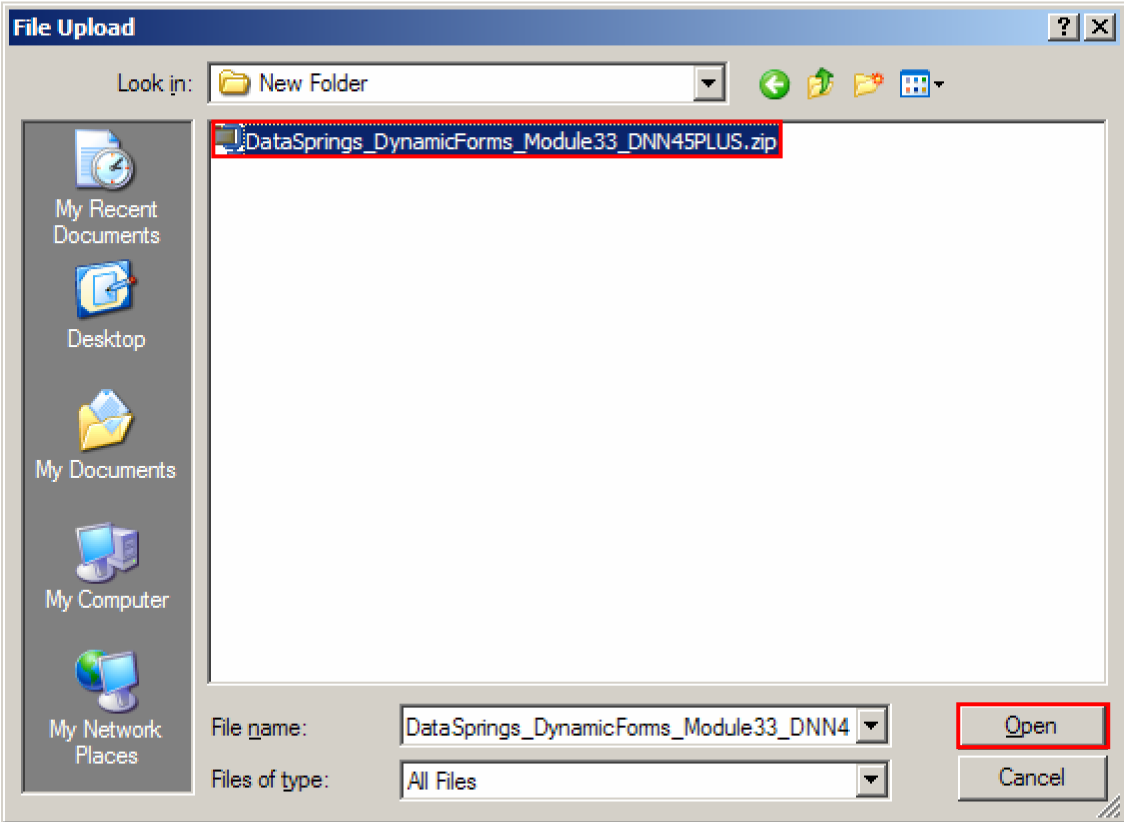


Figure 4: Installation procedure (step 4/6)

Locate the “**DataSprings_DynamicForms_Module33_DNN45PLUS.zip**” and click on the “Open” button. The following screen will be displayed.

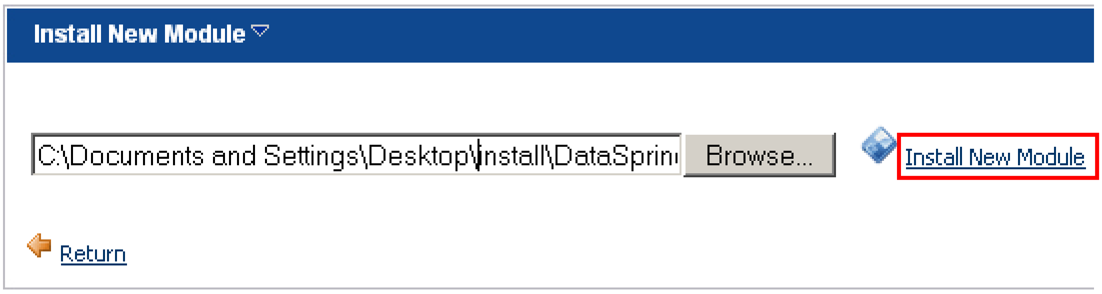


Figure 5: Installation procedure (step 5/6)

Click on the “Install New Module” link. The installation will begin and in couple of moments the screen informing you on successful completion will be displayed.

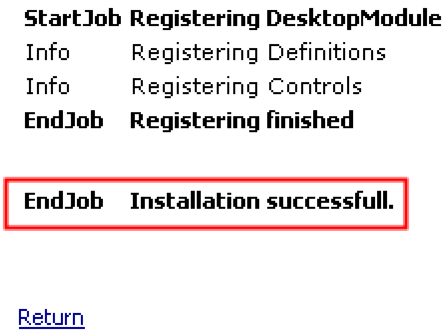


Figure 6: Installation procedure (step 6/6)

Note: please keep track of any errors that appear during the installation. These errors can be helpful if your module has problems.

3 Adding Dynamic Forms Module to a Page

In order to add “Dynamic Forms” module to a desired page follow these steps:

1. Select the “Add New Module” radio button
2. Choose “Dynamic Forms” from the “Module” pull down menu
3. Click on the “Add” link (or the arrow icon).

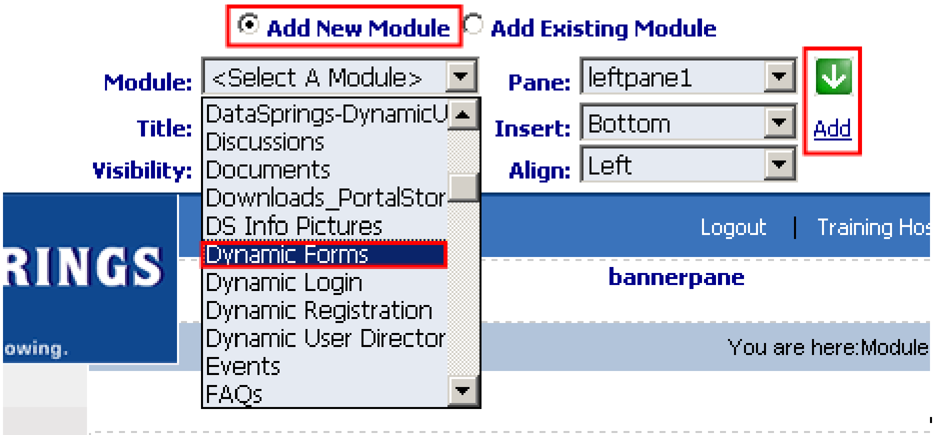


Figure 7: Adding a module to a page

The “Dynamic Forms” module will be added to the page. Click on the arrow next to the title of the module in order to open the main menu (**note**: see section 4 for further information on main menu options).

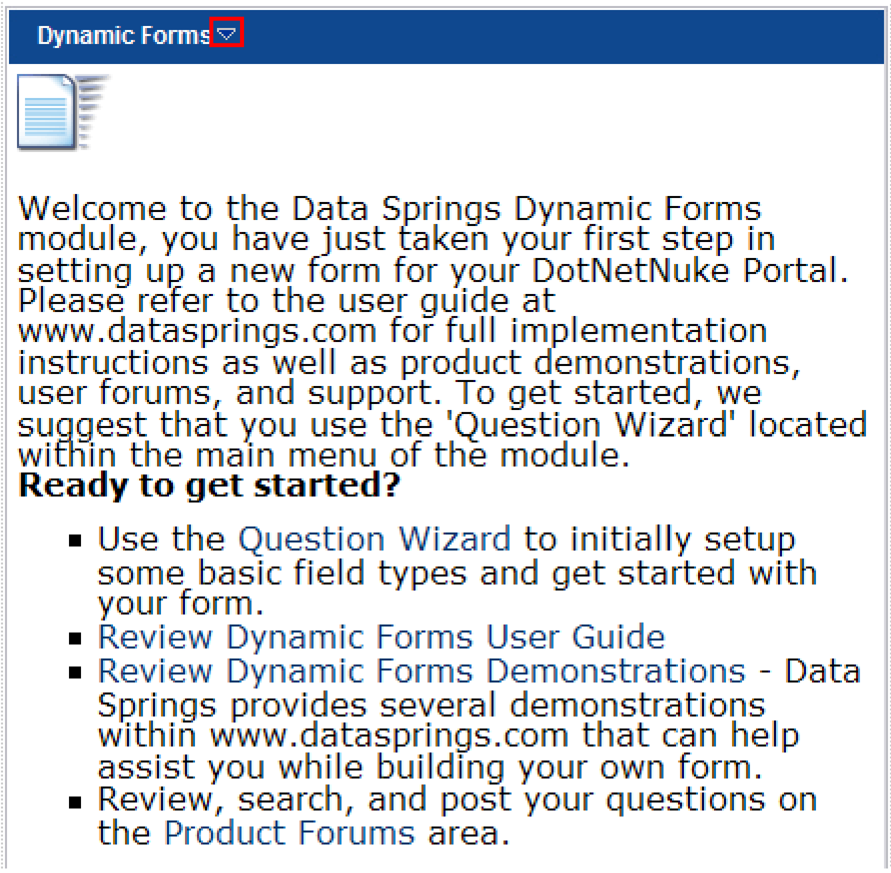


Figure 8: Opening the module main menu.

4 DYNAMIC FORMS INTERFACE

This section of the document will describe the options available within the Dynamic Forms interface.

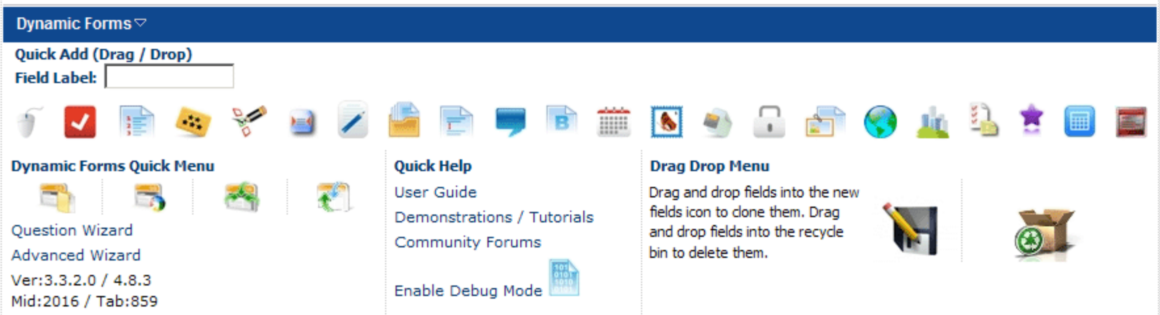


Figure 9: Options within the Dynamic Forms Interface

The following options are available:

- **Dynamic Forms Quick Menu** – this is the quick menu towards the most important functionalities within the Dynamic Forms menu:
    -  - the option for managing the dynamic fields
    -  - the option for managing the module configuration
    -  - the option for managing the completion events
    -  - the option for managing the question events
- **Quick Help** – the part of the interface containing helpful information and instructions on how to use the Dynamic Forms module:
    - **User Guide** – click this option to open the user guide
    - **Demonstrations/Tutorials** – click this option to view the page with the demonstrations and tutorials
    - **Community Forums** – click this option to visit the community forums
    - **Enable Debug Mode** – click this option to enable the debug mode which will display debug information (feedback on errors) which you can use in order to eliminate bugs potential problems
- **Quick Edit** – use the quick edit  option to manage the field right from within the form (without having to open the control panel) – see section 4.1
- **Drag Drop Menu** – this part of the interface is used for creating copies i.e. clones of the existing fields (see section 4.1)
- **Quick Add (Drag/Drop)** – this part of the interface is used for adding new dynamic fields by choosing the desired icon for the field and using the simple drag and drop method (see section 4.4)
- **Field Label** – this field is used for setting the label of the new dynamic question i.e. the name of the field and should be used in combination with the **Quick Add** functionality
- **Recycle bin** – this part of the interface is used for deleting the desired dynamic fields by simply dragging them to the recycle bin (see section 6.3)

4.1 Editing a field with “Quick Edit” option

The quick edit option allows you to edit a field without having to open the control panel i.e. right from the form. In order to edit a field, click the quick edit icon next to the desired field.

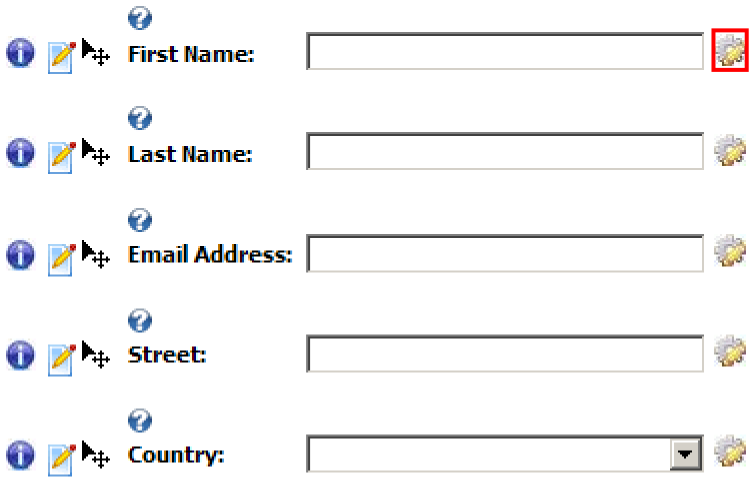


Figure 10: Editing a field by using the "Quick Edit" option (step 1/4)

The “Quick Edit” label will be displayed.

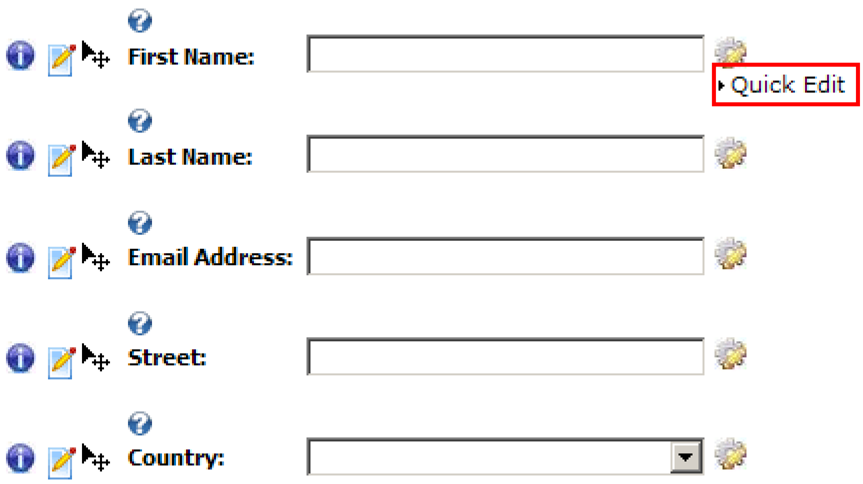


Figure 11: Editing a field by using the "Quick Edit" option (step 2/4)

Place you mouse on the “Quick Edit” label and the menu with further options will be displayed.

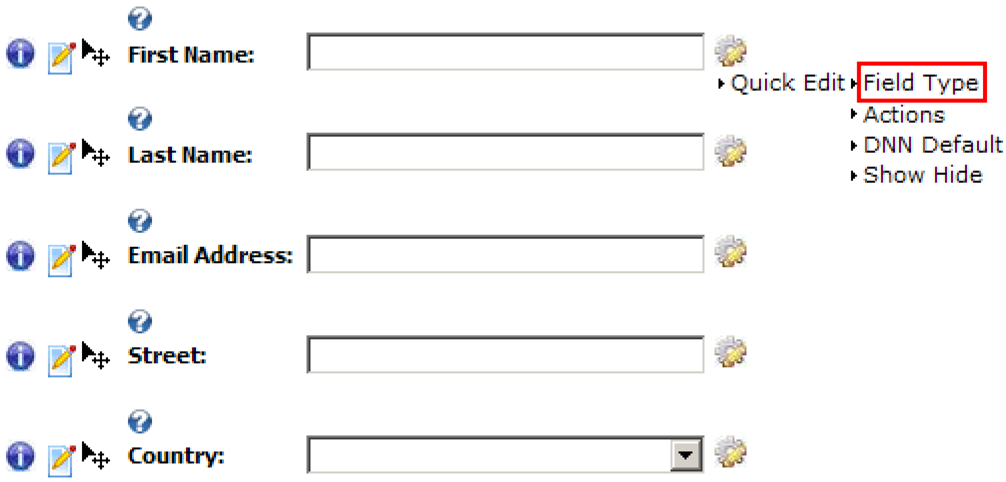


Figure 12: Editing a field by using the "Quick Edit" option (step 3/4)

Place your mouse over the option you wish to edit e.g. “Field Type” to further expand the menu.

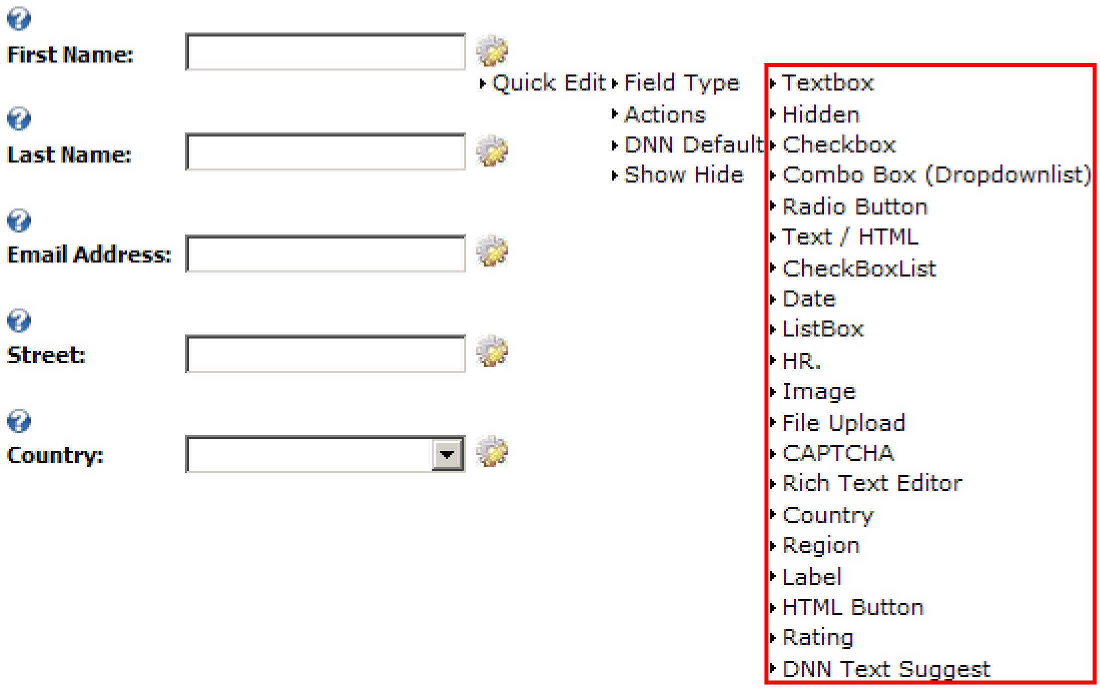


Figure 13: Figure 12: Editing a field by using the "Quick Edit" option (step 4/4)

The menu will give you the option to change the field type by clicking on the desired different type. The change will be immediately visible.

4.2 Creating the copies of the existing fields

In order to quickly create a copy i.e. a clone of the existing fields directly from within the interface, click on this icon next to the desired field and move it over to the Drag Drop Menu icon

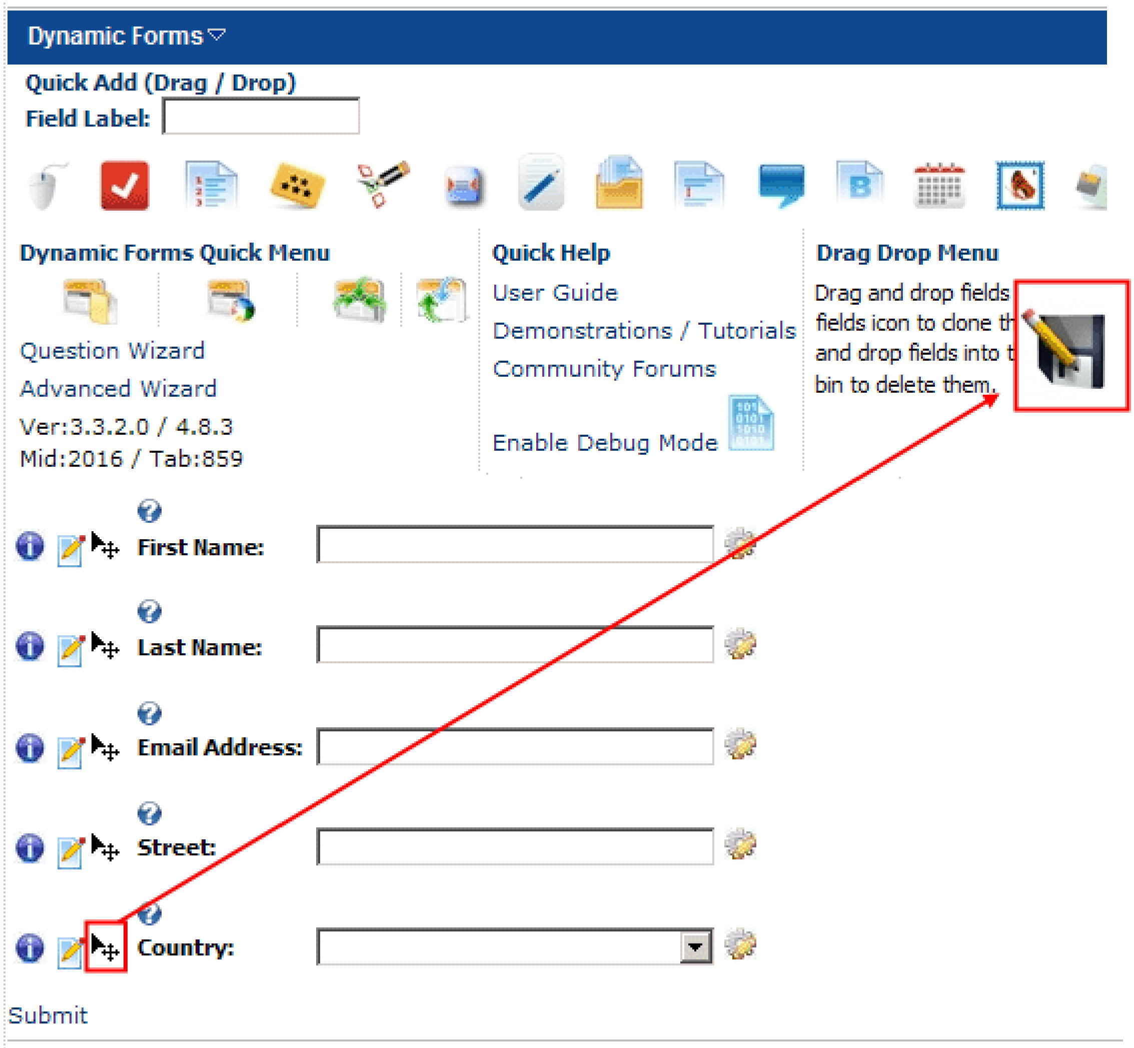


Figure 14: Cloning an existing field (step 1/2)

Once you release the icon on the drag drop menu icon, the screen will be refreshed containing the newly created clone of the field.

Figure 15: Cloning an existing field (step 2/2)

Use the edit icon next to the field to further adjust the field if necessary.

4.3 Viewing field info

In order to view quick information about the form field, place your mouse over the info icon next to the desired field.

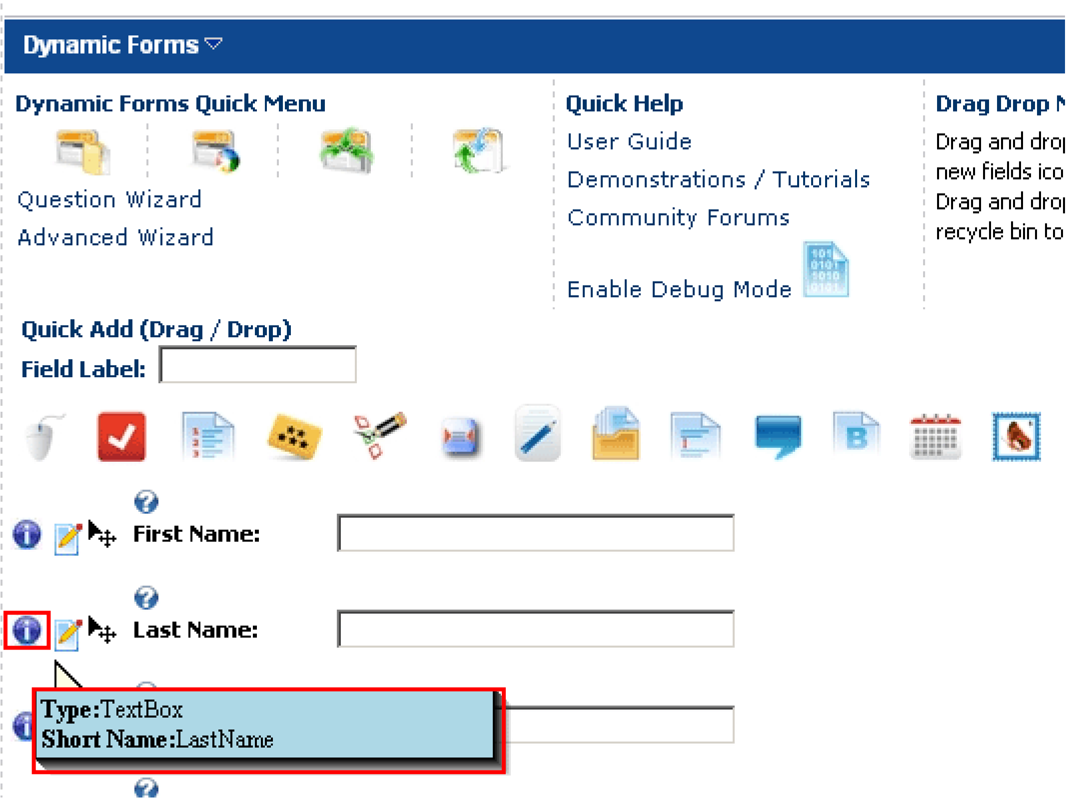


Figure 16: Viewing the field info

The cloud-like blue layer will be displayed containing information about the type of the field as well as the short name for the field. Additional enabled features will also be displayed, such as if the field is required or if advanced features are enabled to initially hide the field or pass the field value via a querystring or session variable.

4.4 Using the Quick Add functionality

The Quick Add functionality is a simple new way to add new dynamic fields directly from the interface.

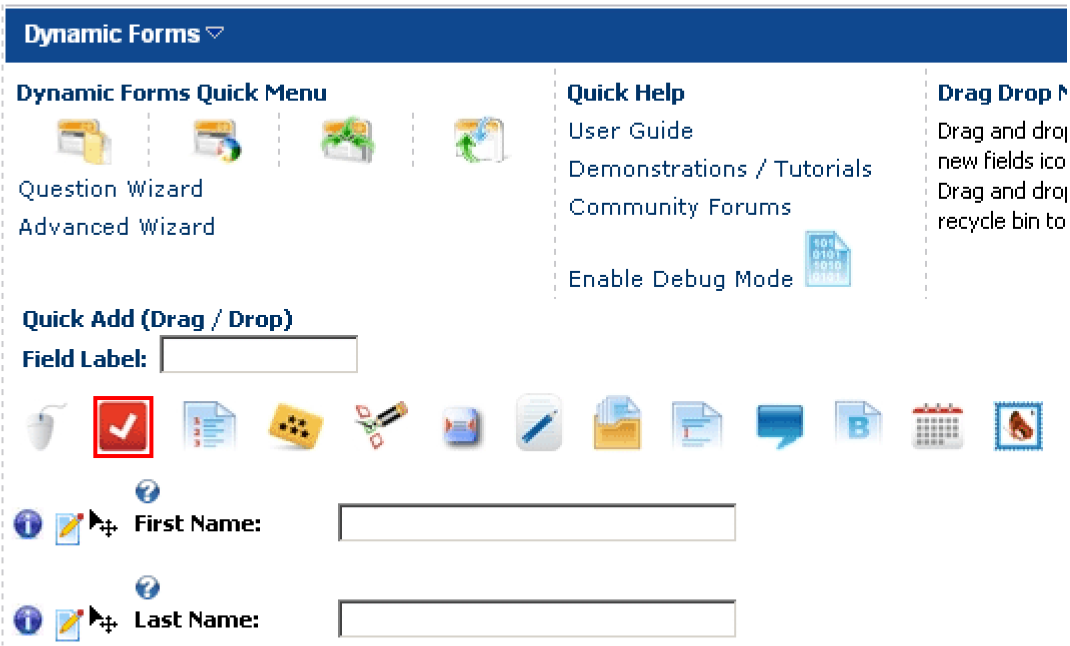


Figure 17: Selecting the desired icon/dynamic field

In order to quickly add a new dynamic field, click on one of the shortcut icons and then drag&drop within the desired location in the form. Please note that the location where you release the icon will determine the sort order of the new dynamic field.

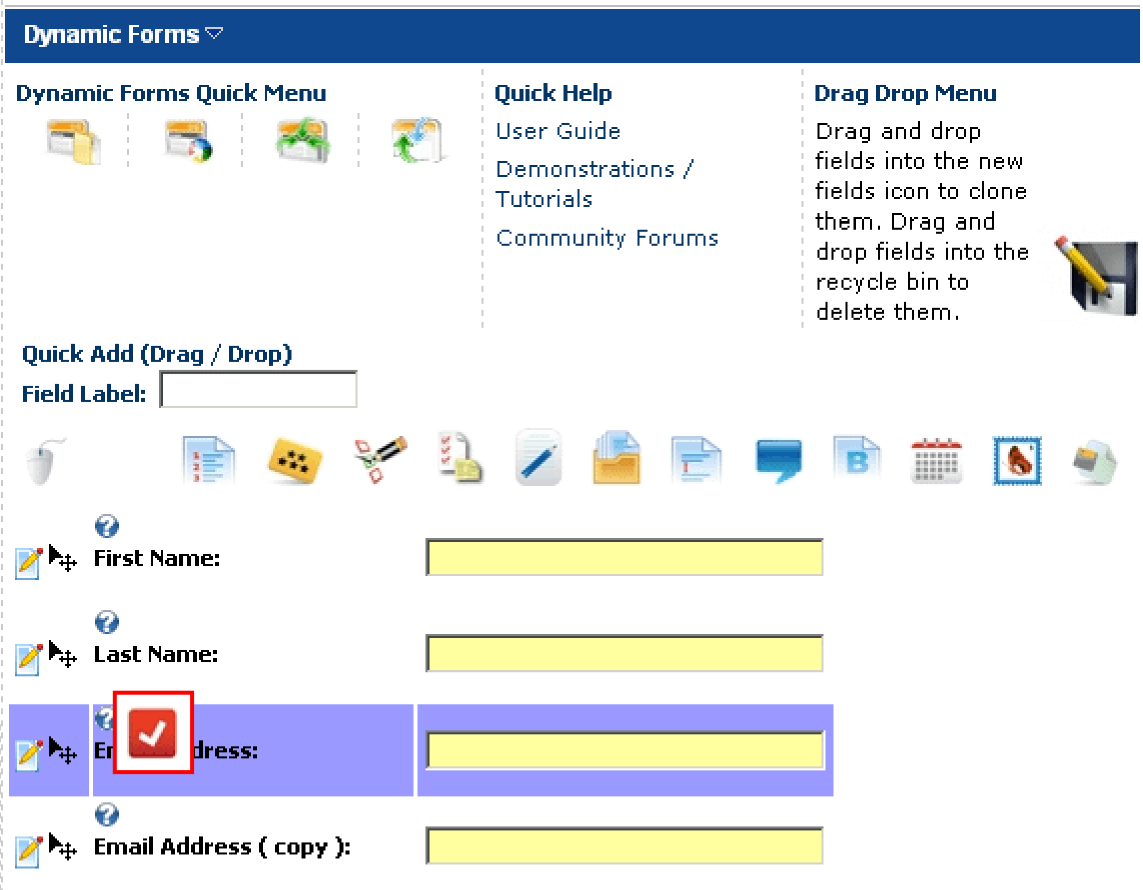


Figure 18: Releasing the new field

Once you release the icon, the new field will be added to the form.

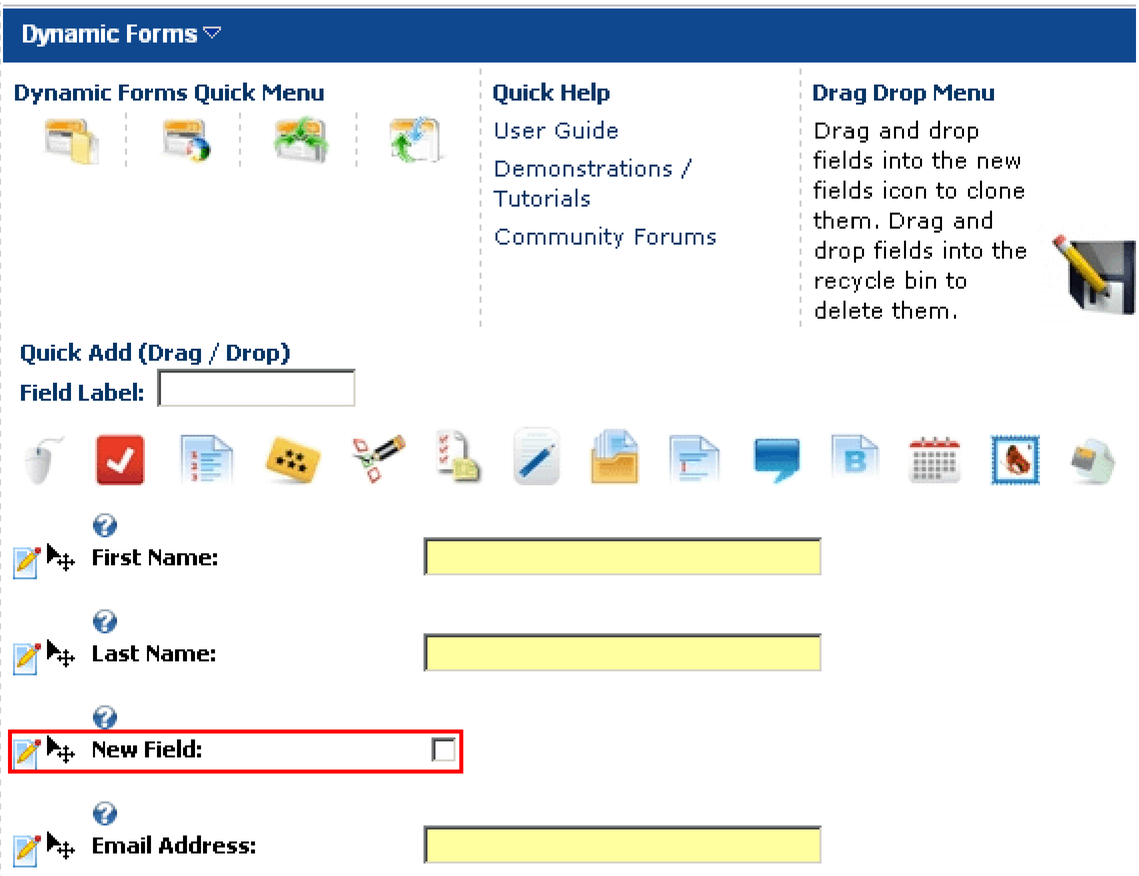


Figure 19: The new field successfully added to the form

Once the new field has been added, you can place the mouse pointer over it until you see the edit field icon. Click on this icon and you will be able to set the name of the field.

- **Note:** you can also specify the field label before you drag and drop the icon onto the form. You can specify this in the ‘Field Label’ area above the new field icons (top left hand side and below the quick drag drop menu) – see section 4.5.

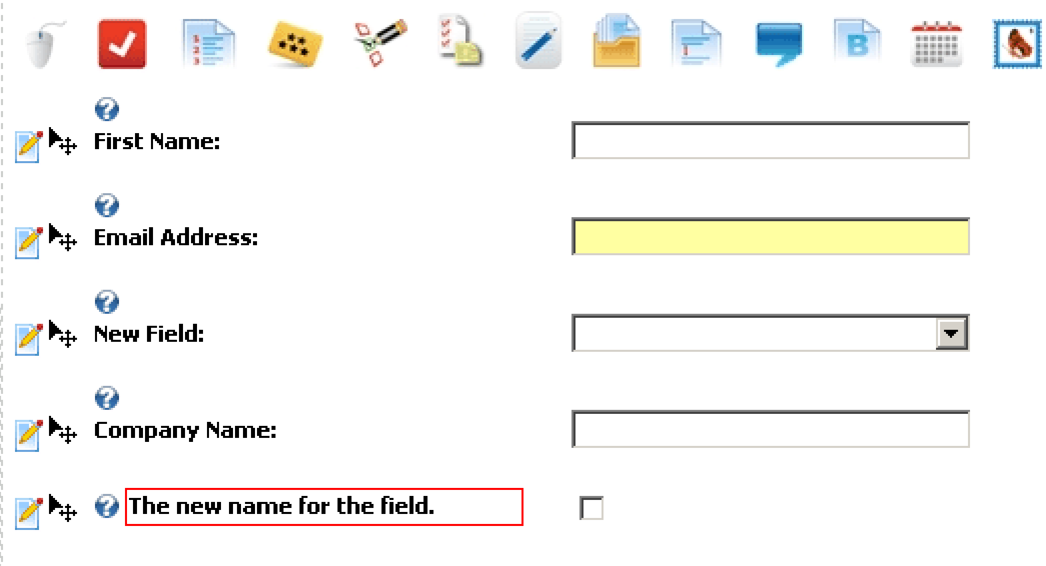


Figure 20: Setting the new name for the field

For all other adjustments of the field use this icon which will open the usual edit page with all available parameters for the field.

The following icons are available within the interface

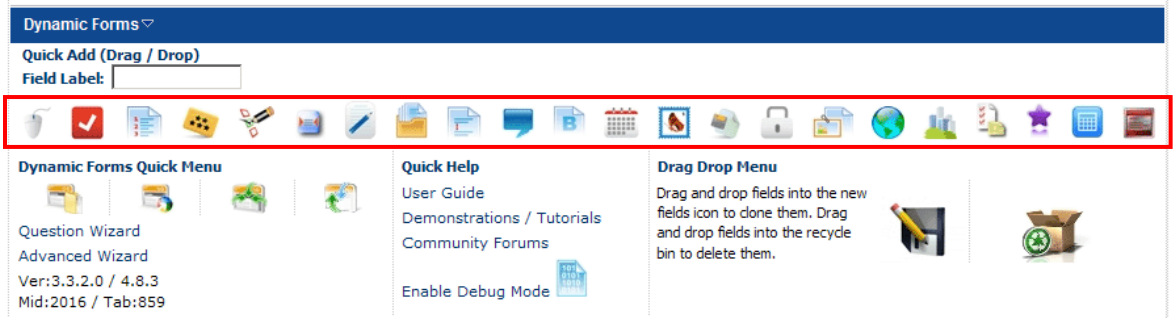


Figure 21: Available icons for adding new dynamic fields

-  - the option for adding a new **textbox**
-  - the option for adding a new **checkbox** field
-  - the option for adding a new **combo box**
-  - the option for adding a new **radio button** field
-  - the option for adding a new **checkbox list** field
-  - the option for adding a new **listbox** field
-  - the option for adding a new **text/HTML** field
-  - the option for creating a new **hidden field**
-  - the option for creating an **HR ruler**
-  - the option for creating a new **label field**
-  - the option for creating a new **HTML input button field**
-  - the option for creating a new **date field**
-  - the option for creating a new **image** field
-  - the option for creating a new **file attachment** field
-  - the option for creating a new **CAPTCHA** field
-  - the option for creating a new **rich text box** field
-  - the option for creating a new **DNN® country** field
-  - the option for creating a new **DNN® region** field
-  - the option for adding a new **gridview** field
-  - the option for adding a new **Rating (AJAX)** field

- - the option for adding a new **Data Grid** field
- - the "**DNN Text Suggest"** field (this functionality suggests the text as the user types recognizing the word/field from the database and based on the SQL query to the database – see section 7.16)

4.5 Setting the field name – using the Field Label functionality

The purpose of the "Field Label" functionality is to allow you to quickly and easily set the name of the new field by entering its name into the "Field Label" input field.

For example, if you want to add a "Last Name" field to your form, enter the "Last Name" text into the "Field Label" input field and then drag and drop the desired field icon onto the form.

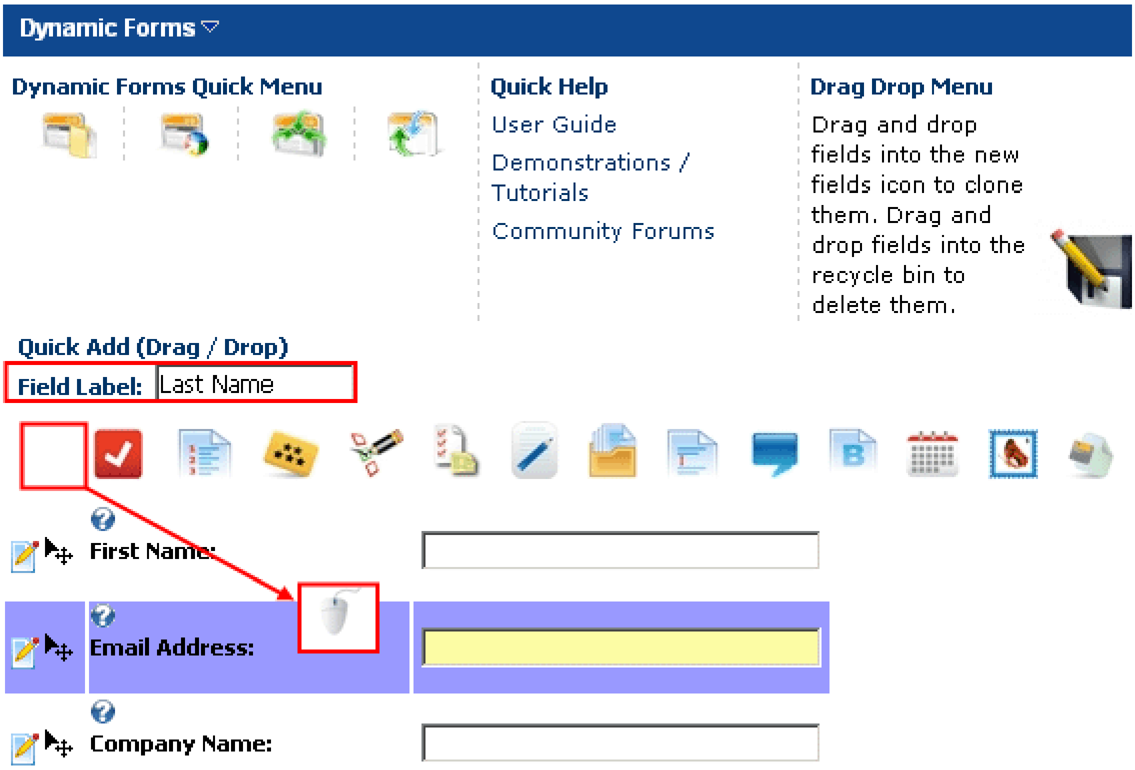


Figure 22: Setting the field name (step 1/2)

Once you release the field, the page will be refreshed containing the new field which is already properly named.

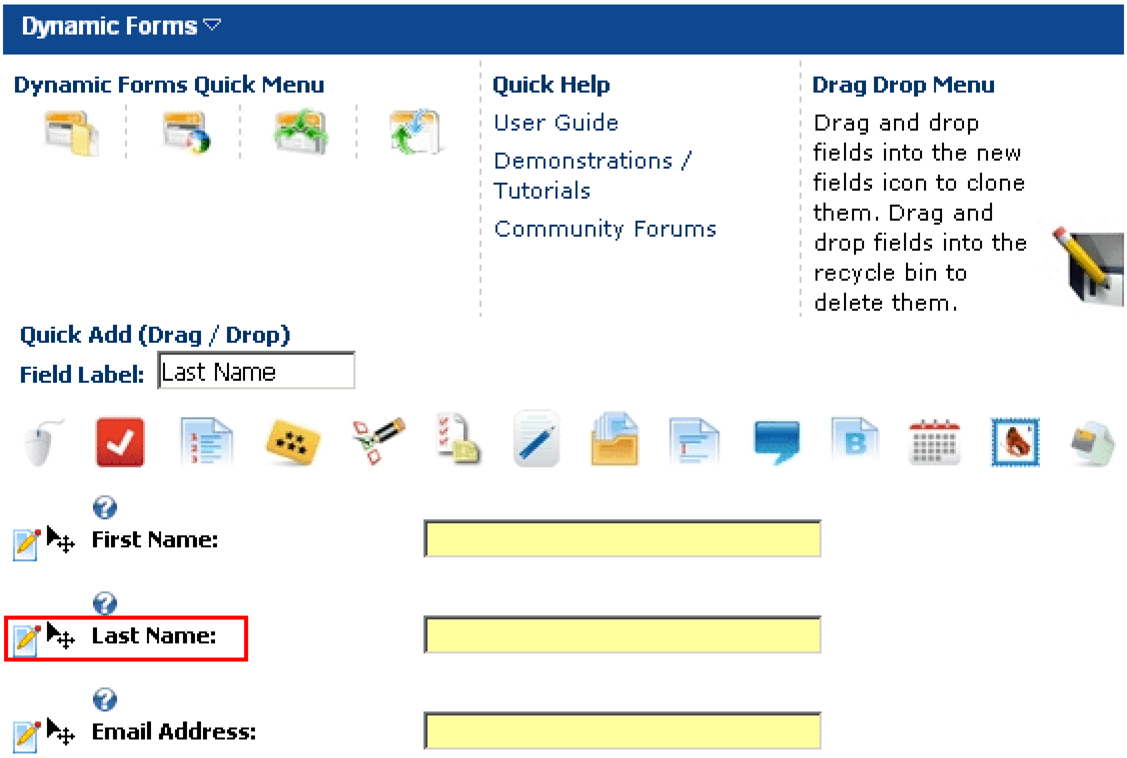


Figure 23: Setting the field name (step 2/2)

4.6 Moving dynamic questions to the Recycle Bin

The “Recycle Bin” functionality is used the same way as your Windows Recycle bin. The purpose of this functionality is to allow you to remove the items from your dynamic form as simply as possible, but still offer a possibility to restore them at any time (**note:** unless removed from the recycle bin altogether).

In order to move the desired item to the recycle bin, click on this icon next to the item and move it towards the recycle bin icon .

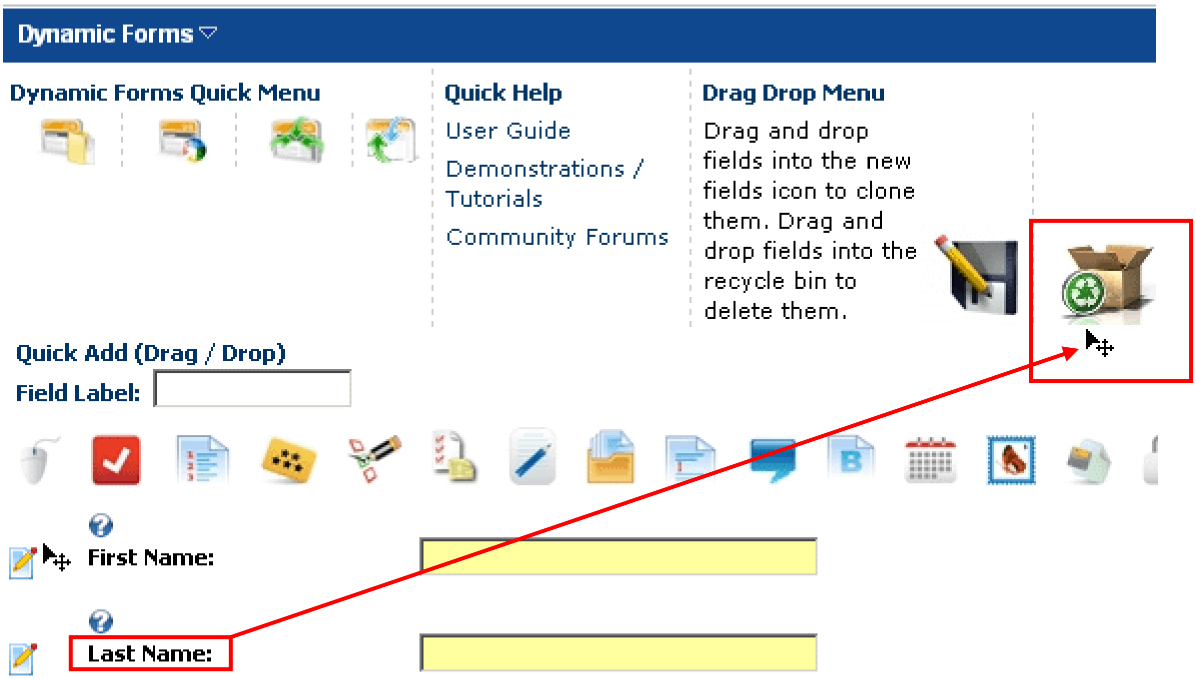


Figure 24: Moving a dynamic question to a recycle bin

The desired item will be removed from the dynamic form and placed into the recycle bin. The items in the recycle bin can be restored or permanently deleted.

In order to access the contents of the recycle bin, choose the “Recycle Bin” option from the “Advanced Features” menu (see section 6.3).

5 DYNAMIC FORMS MAIN MENU

This section of the document will give the definition of the “Dynamic Forms” main menu options. In order to start using the main menu, click on the arrow next to the title “Dynamic Forms”.

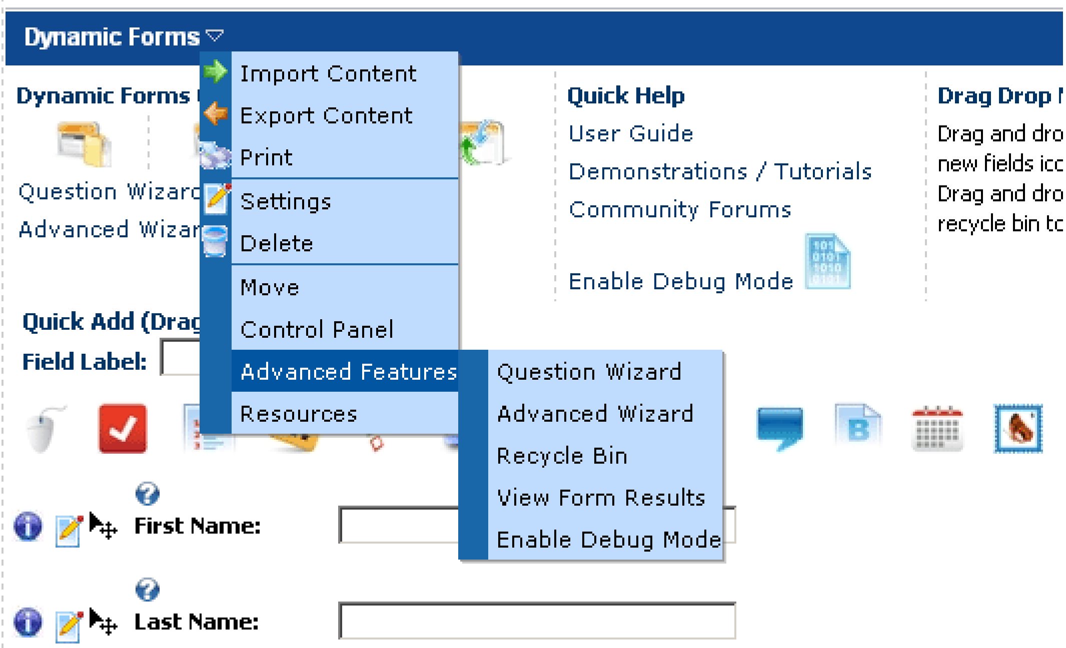


Figure 25: Opening the main menu

The following options are available inside this screen:

- **Import Content** – option for importing content (see section 8)
- **Export Content** – option for exporting content (see section 9.1)
- **Print** – option for printing the contents of the page
- **Settings** – option for managing settings (see section 10)
- **Delete** – option for deleting a module (see section 11)
- **Move** – option for placing the module in the desired part of the page:
    - Move To leftpane
    - Move To bannerpane
    - Move To contentpane
    - Move To rightpane
    - Move To bottompane
    - Move To bottompane1
    - Move To bottompane2
- **Control Panel** – the option for accessing the control panel (see section 7
- **Advanced Features** – option for managing the advanced module features (see section 6)
    - **Question Wizard** – the wizard for creating the most common set of questions fast (e.g. “first name”, “last name”, “email”, etc.) – see section
    - **Advanced Wizard** – the wizard for creating advanced questions (see section 6.2)
    - **Recycle Bin** – the option for accessing and managing the recycle bin with the deleted items (see section 6.3)

- **View Form Results** – the option for viewing the form results (see section **8**)
- **Enable Debug Mode** – the option for enabling the debug mode which will give you feedback about the possible errors with the module so you could solve the potential issues more easily

* **Resources** – this submenu contains the following options:
  - User Guide
  - Community Forums
  - Demonstrations/Tutorials

6 MANAGING THE ADVANCED MODULE FEATURES

In order to start managing the advanced module features, choose one of the available options from the "Advanced Features" menu.

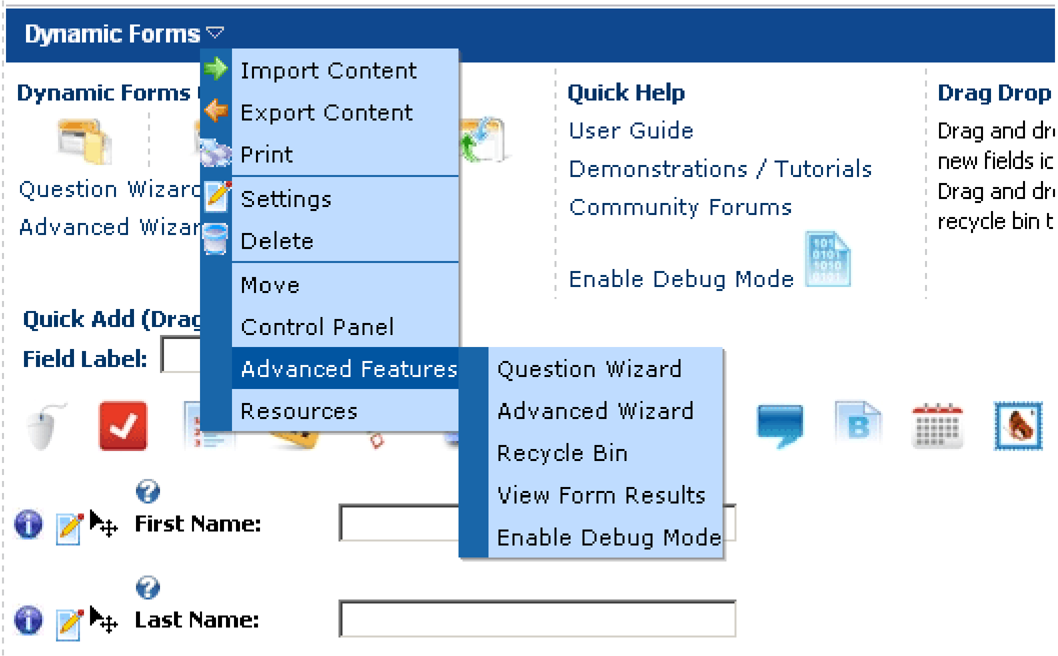


Figure 26: Managing the advanced features

The following options are available within the "Advanced Features" menu:

- **Question Wizard** – the option for using the question wizard, a program designed to help you create the most common questions as quickly as possible (see section _)
- **Advanced Wizard** – the option for using the advanced wizard (see section 6.2)
- **Recycle Bin** – the option for using the recycle bin (see section 6.3)
- **View Form Results** – the option for viewing the form results (see section 8)
- **Enable Debug Mode** – the option for enabling the debug mode which will give you feedback about the possible errors with the module so you could solve the potential issues more easily

6.1 Using the Question Wizard

The "Question Wizard" is a very useful and practical tool which can help you create a dynamic form in a matter of seconds. The purpose of the wizard is to help you quickly create the most commonly used questions in a form (**e.g.** "First name" "Last name").

After using the wizard to create the most common questions, you can continue to customize the form further by incorporating other elements/questions according to your preference.

In order to start using the "Question Wizard", choose the "Question Wizard" option from the "Advanced Features" menu.

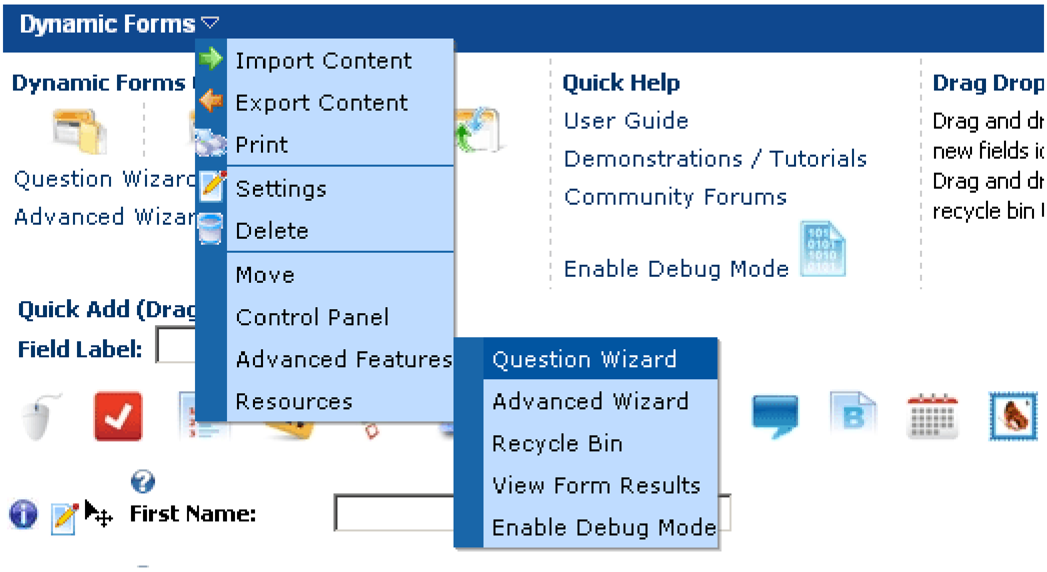


Figure 27: Choosing the "Question Wizard" option

The following screen will be displayed.

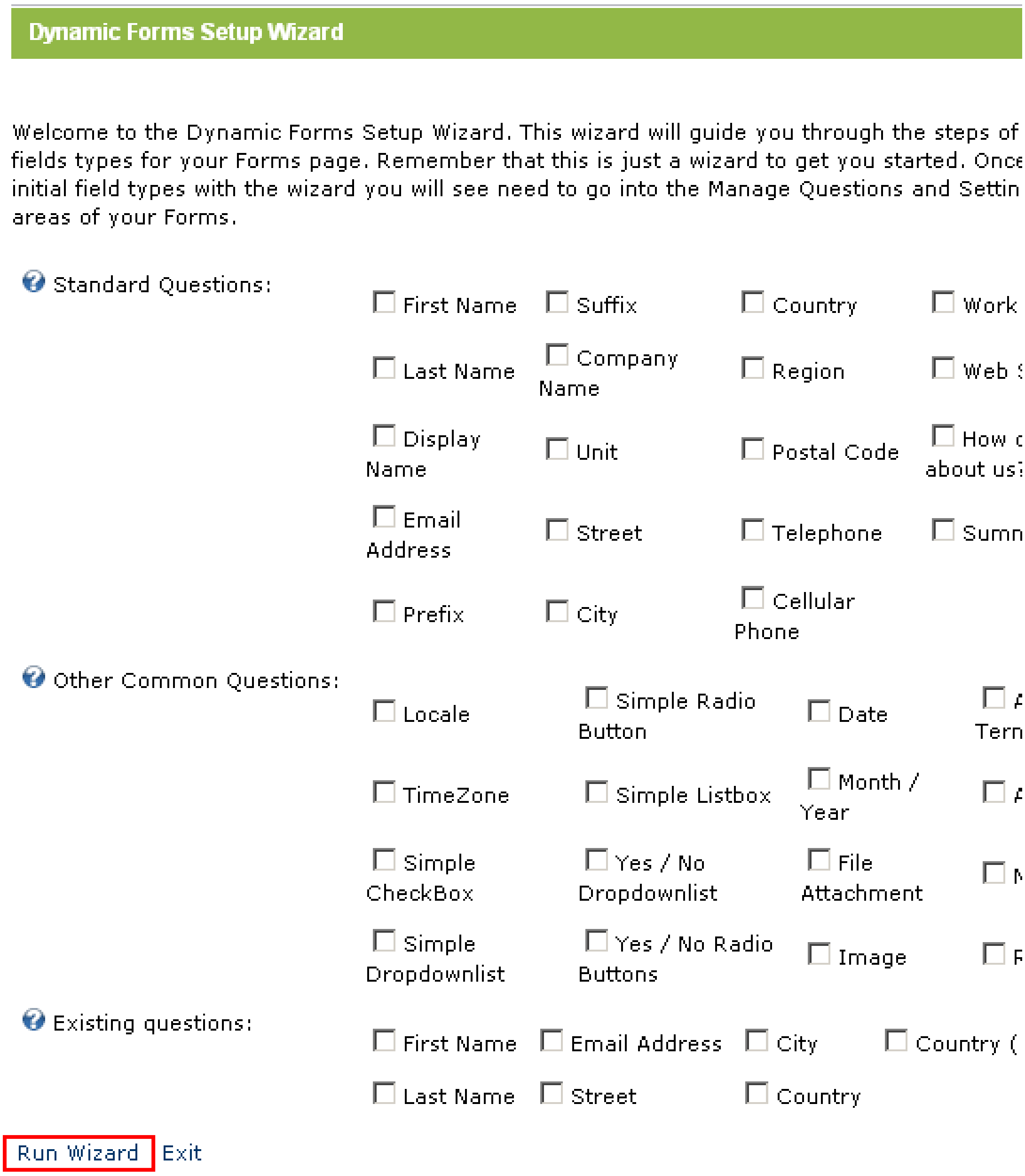


Figure 28: Choosing the desired questions

The questions are presented in 2 categories which are:

- **Standard Question** – this part of the screen contains most widely used questions in a form; just select the questions you wish to include in to your form by selecting the checkbox next to the desired question
- **Other Common Questions** – this part of the screen contains other questions that are commonly used within a form; select the ones you wish to include in your form
- **Existing Questions** – this part of the screen contains questions that are already created within the form. You can quickly clone these questions and the wizard will create new questions from the existing fields and their properties

Once you select the questions you wish to include in your form, click on the “Run Wizard” link. The form will be generated.

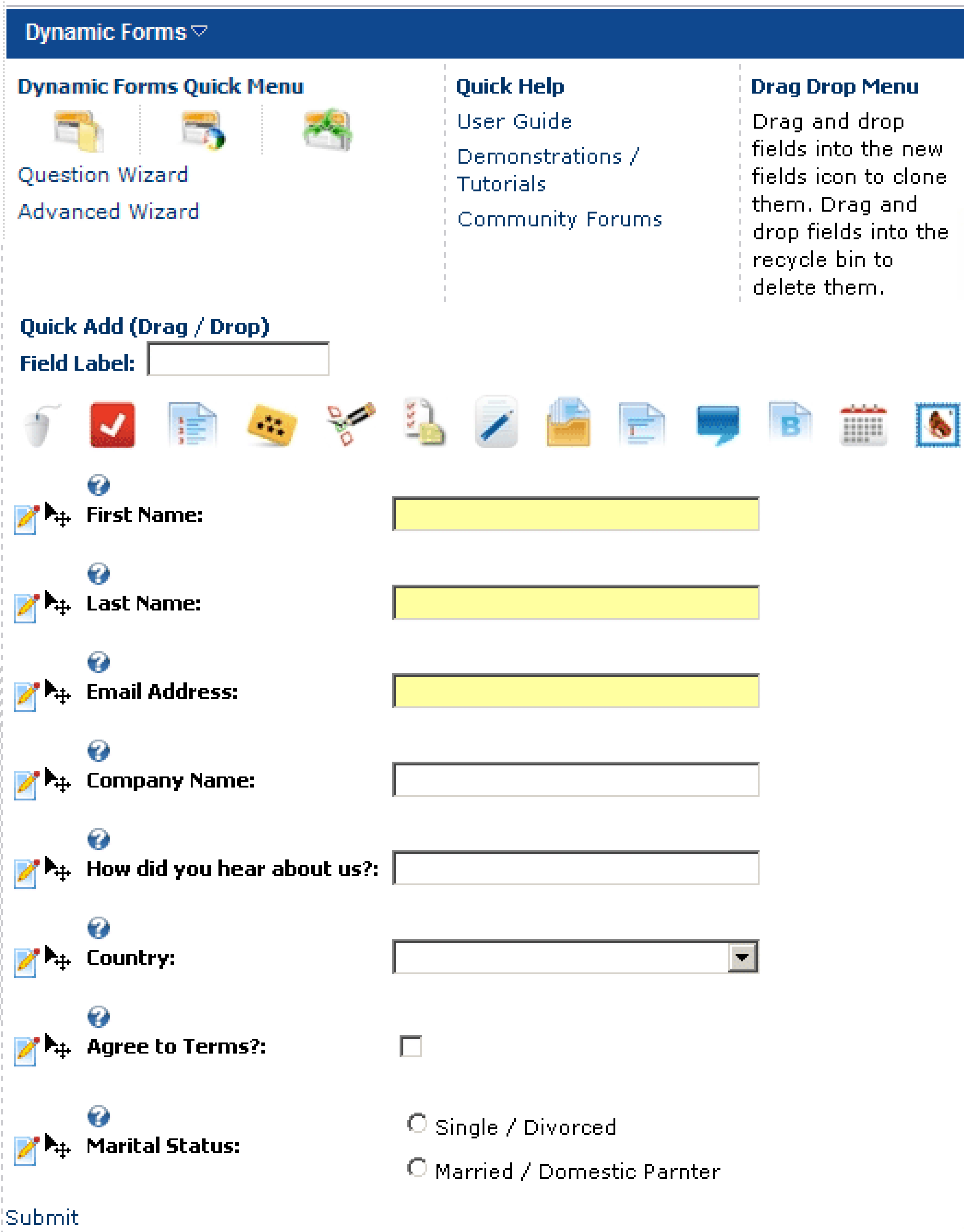


Figure 29: The form successfully created

Note:

- Use the “Dynamic Forms Quick Menu” in order to quickly access important program options
  - o - option for managing the dynamic fields
  - o - option for setting up the module configuration
  - o - option for setting up completion events
  - o - option for setting up events initiated while the user is filling out the form
- You can use the edit icon next to each of the questions in order to open the page with the question details where you can modify it

- See section 7.1 for further information about the available form elements as well as instructions on how to fully customize your form
- **Question wizard** – see section _
- **Advanced wizard** – the tool designed to help you manage and maintain properties for multiple fields as simply as possible (see section 6.2)

6.2 Using the Advanced Wizard

The "Advanced Wizard" is a tool designed to help you manage and maintain properties for multiple fields as simply as possible. For example, you can use this wizard to make multiple fields in your form mandatory, by simply selecting those fields and applying the appropriate "Question Action Type" (in this case "Required Field").

Then use the wizard again to specify the required field text, required field error message, or any of the other actions from within the "Question Action Type" pull down menu.

In order to start using the "Advanced Wizard", choose that option from the "Advanced Features" menu.

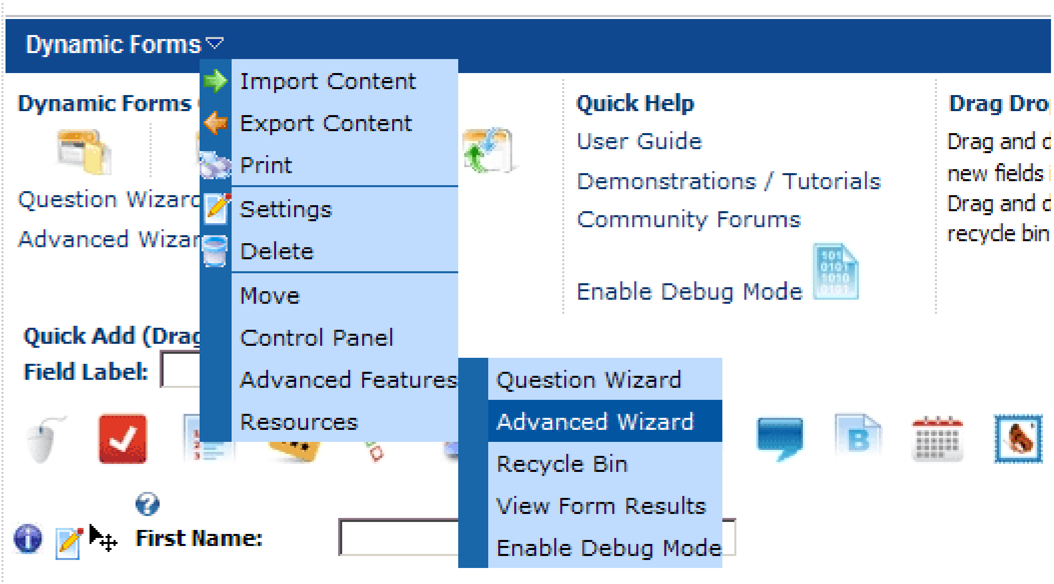


Figure 30: Choosing the "Advanced Wizard" option

The following screen will be displayed.

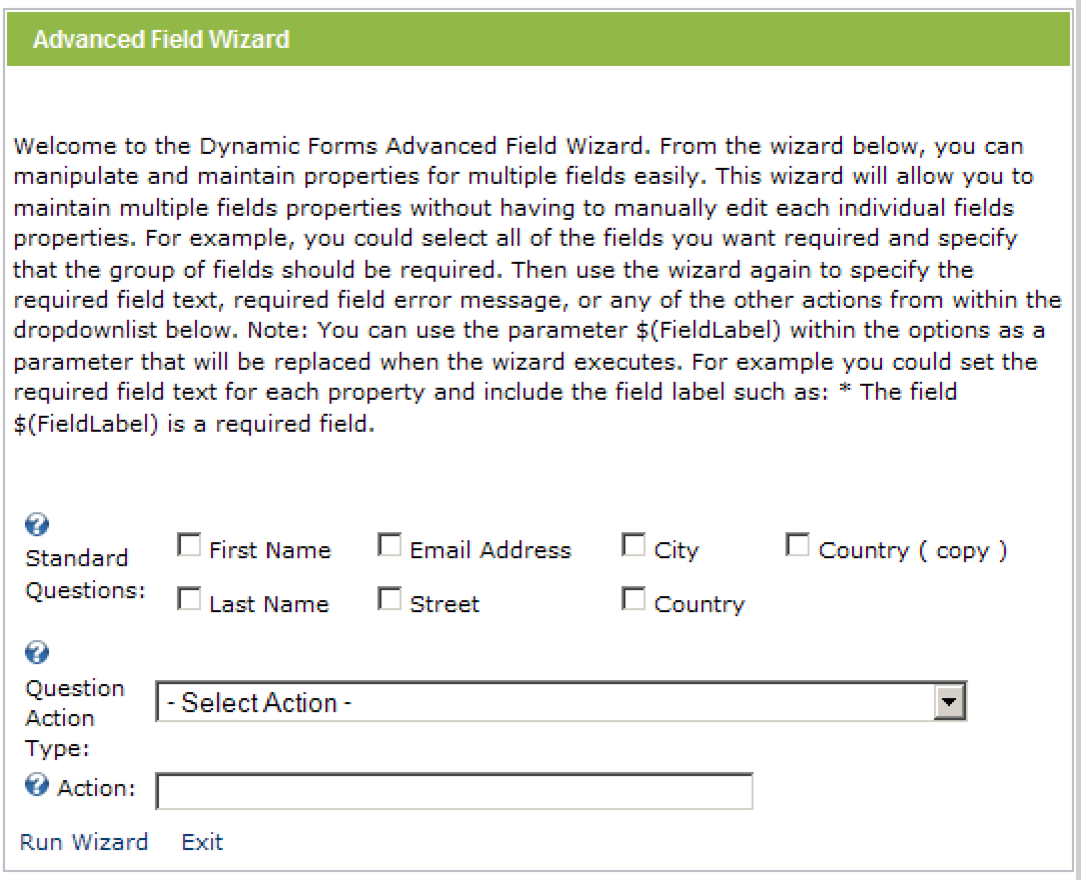


Figure 31: Using the "Advanced Wizard"

The following options and parameters are available:

- **Standard Questions** – select the question you wish to apply the rest of the options to
- **Question Action Type** - choose the question action type from this pull down menu
    - **For example**: choose the “Hide question from these roles” in case you wish to prevent the question from being displayed to specific roles on your system

The following options are available within the pull down menu:

- Help Text
- Default Value
- Example Text
- Client Side Event
- Hide Question from these roles
- Hide from anonymous users
- Hide until forced visible by question event
- Pass values of this question to querystring variable
- Retrieve values from querystring variable for this question
- Pass values of this question to session variable
- Retrieve values from session variable for this question
- Pass values of this question to cookie variable
- Retrieve values from cookie variable for this question
- Do not display the field results within the field results module
- Do not save results to the database

- Label class
- Field class
- Column span
- Row span
- Field width
- Add field to same row as previous question
- Display label and field in the same column
- Show help text as popup text
- Pop up text width
- Text box max length
- Required field
- Required field error text
- Required field error message
- Enable regular expression
- Regular expression
- Regular expression error message
- Use SQL options for field
- SQL query for field options

- **Action** – this part of the screen will display the available action based on the choice made within the “Question Action Type” pull down menu (**e.g.** in case you choose to hide the question from specific roles, this part of the screen will list all available roles so you could select those you wish to hide the question for)
- **Run Wizard** – click on this link to start the wizard and execute the associated action

6.3 Using the Recycle Bin

The “Recycle Bin” option functionality is used the same way as your Windows recycle bin. It is a storage for unwanted files which you can decide to restore or permanently delete at any time.

In order to start managing the recycle bin, choose option “Recycle Bin” from the “Advanced Features” menu.

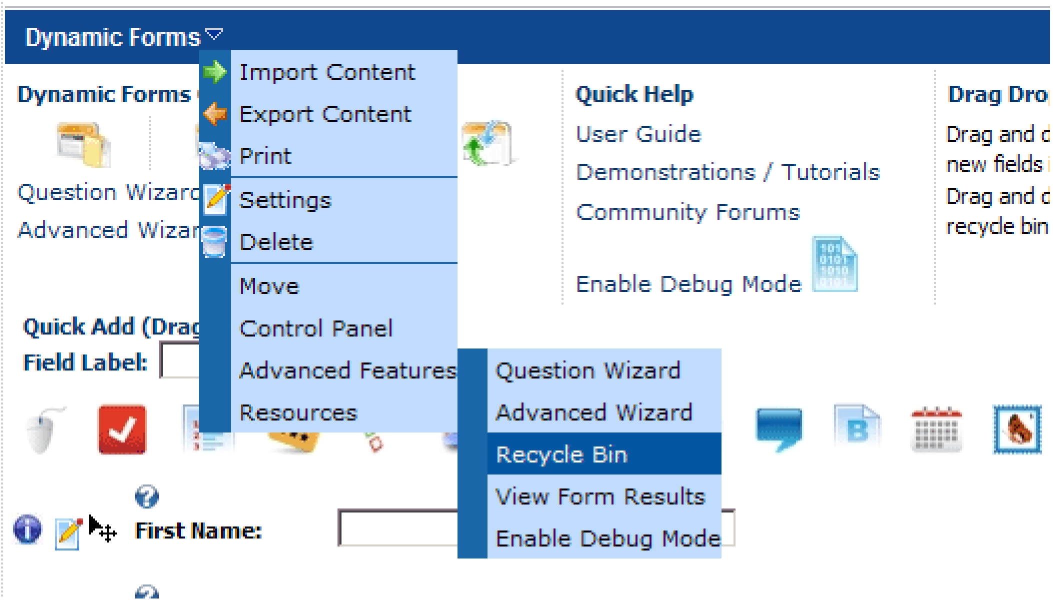


Figure 32: Choosing the "Recycle Bin" option

The following screen will be displayed.

Figure 33: The contents of the Recycle Bin

The items inside the Recycle bin are organized in the following categories:

- Form Questions
- Completion Events
- Question Events

The following options are available:

- **Restore Objects/Exit** – select the items you wish to restore and click on this link to restore them and leave the recycle bin
- **Empty Recycle Bin/Exit** – click on this link to permanently delete all items inside the recycle bin
- **Exit** – click on this link to exit the recycle bin

7 USING THE CONTROL PANEL

In order to start using the option available inside the control panel, choose that option from the main menu.

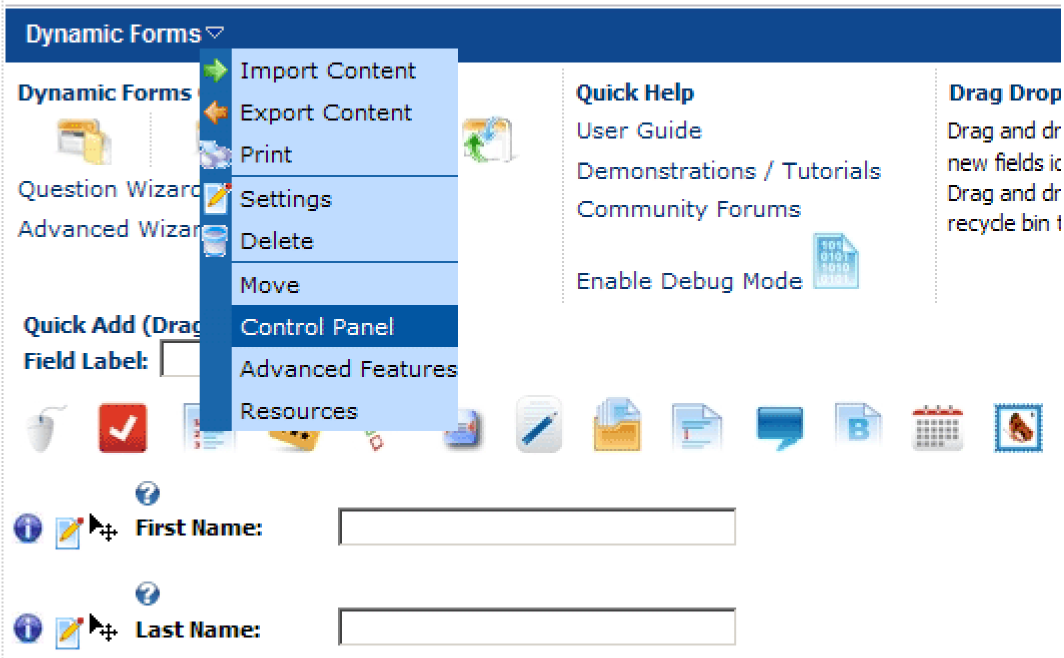


Figure 34: Accessing the control panel

The following page will be displayed.

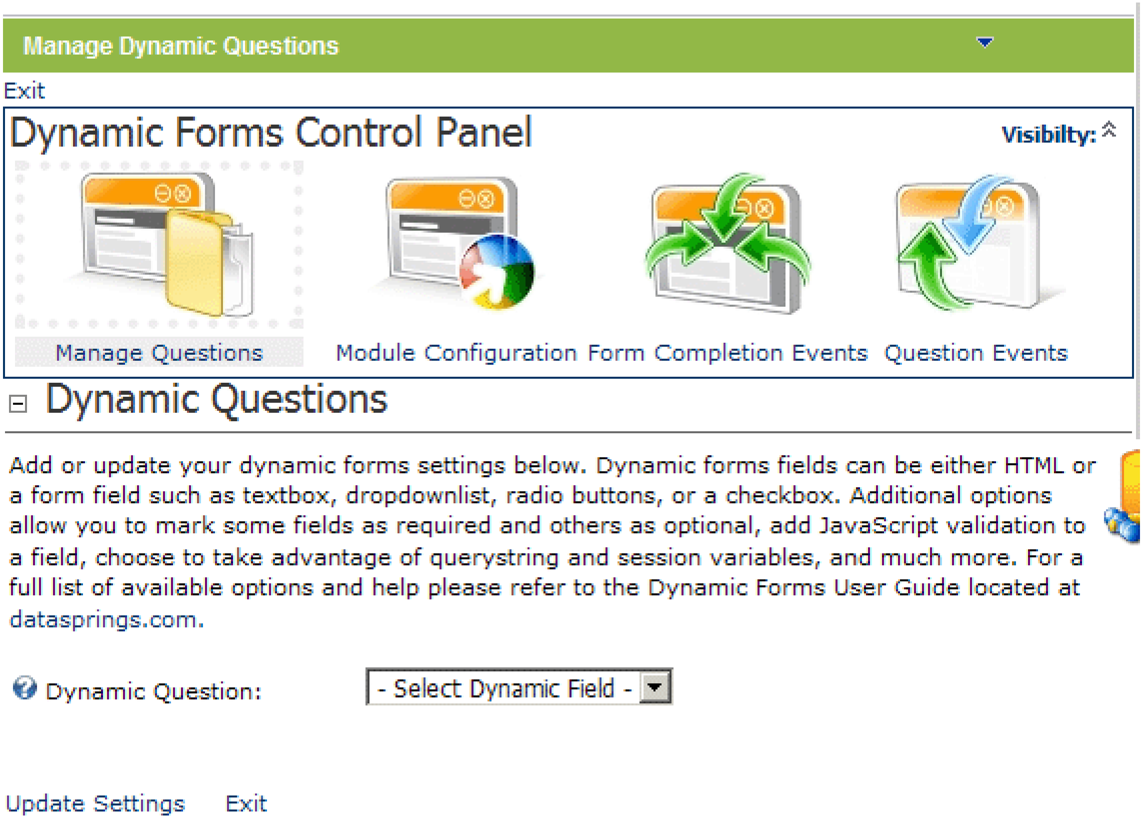


Figure 35: The control panel options

The following options are available:

- **Manage Questions** – option for managing the dynamic questions (see section 7.1)
- **Module Configuration** – option for managing module settings, general configuration, form layout, validation settings, payment gateway, and other general features (see section 7.32)
- **Form Completion Events** – option for managing the form completion events (see section 7.50)
- **Question Events** – option for managing question events (see section 7.53)

7.1 Managing Questions/Settings

In order to start managing questions and settings, choose option "Control Panel" from the main menu. The page for managing the questions will is displayed by default but you can always access it by clicking the "Manage Questions" link.

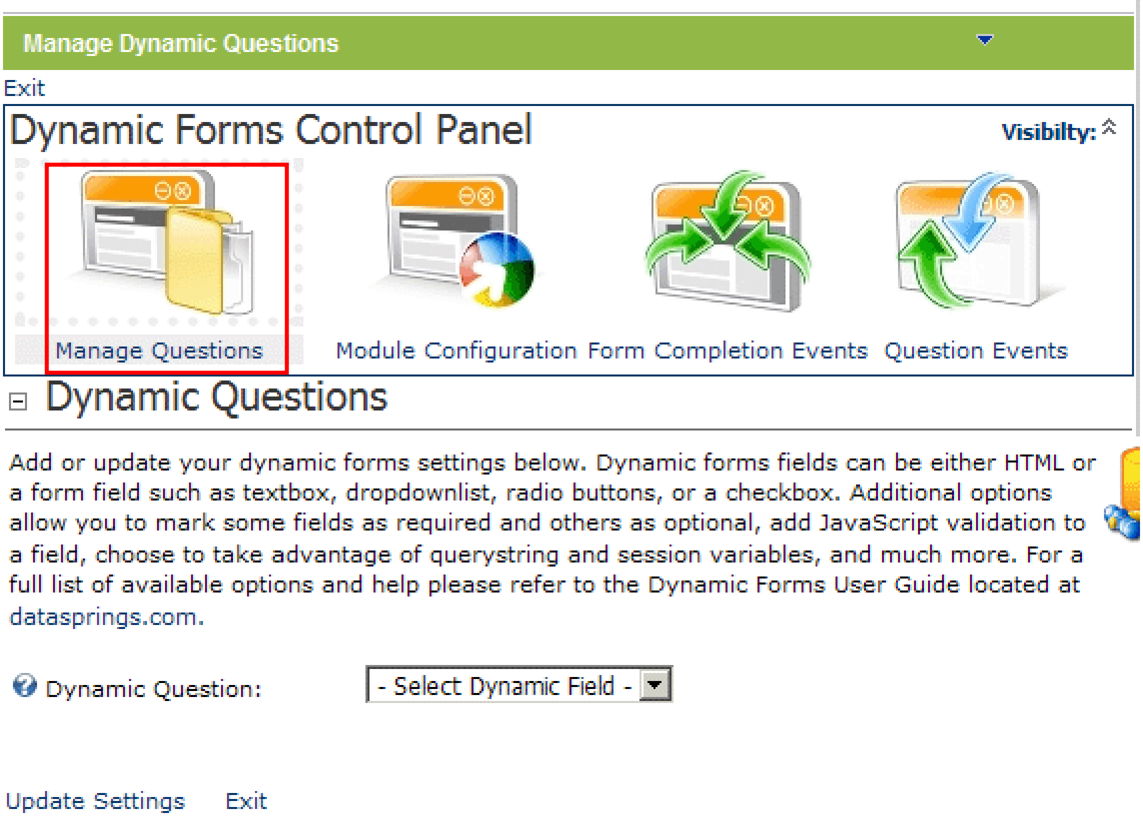


Figure 36: Choosing option "Manage Questions "

The following options are available inside the "Dynamic Question" pull down menu:

- **New Dynamic Field** – option for creating a new dynamic field (see section 7.2)
- **Editing Existing Dynamic Field** – in order to edit an existing field, choose the desired field from this pull down menu (see section 7.30)
- **Deleting a Dynamic Field** – in order to delete a field, select the desired field from this list and click on the "Delete" link in the screen displayed next (see section 7.31)

7.2 Creating a new question

In order to create a new question, choose option “New Dynamic Field” from the pull down menu. The screen for setting question parameters will be displayed.

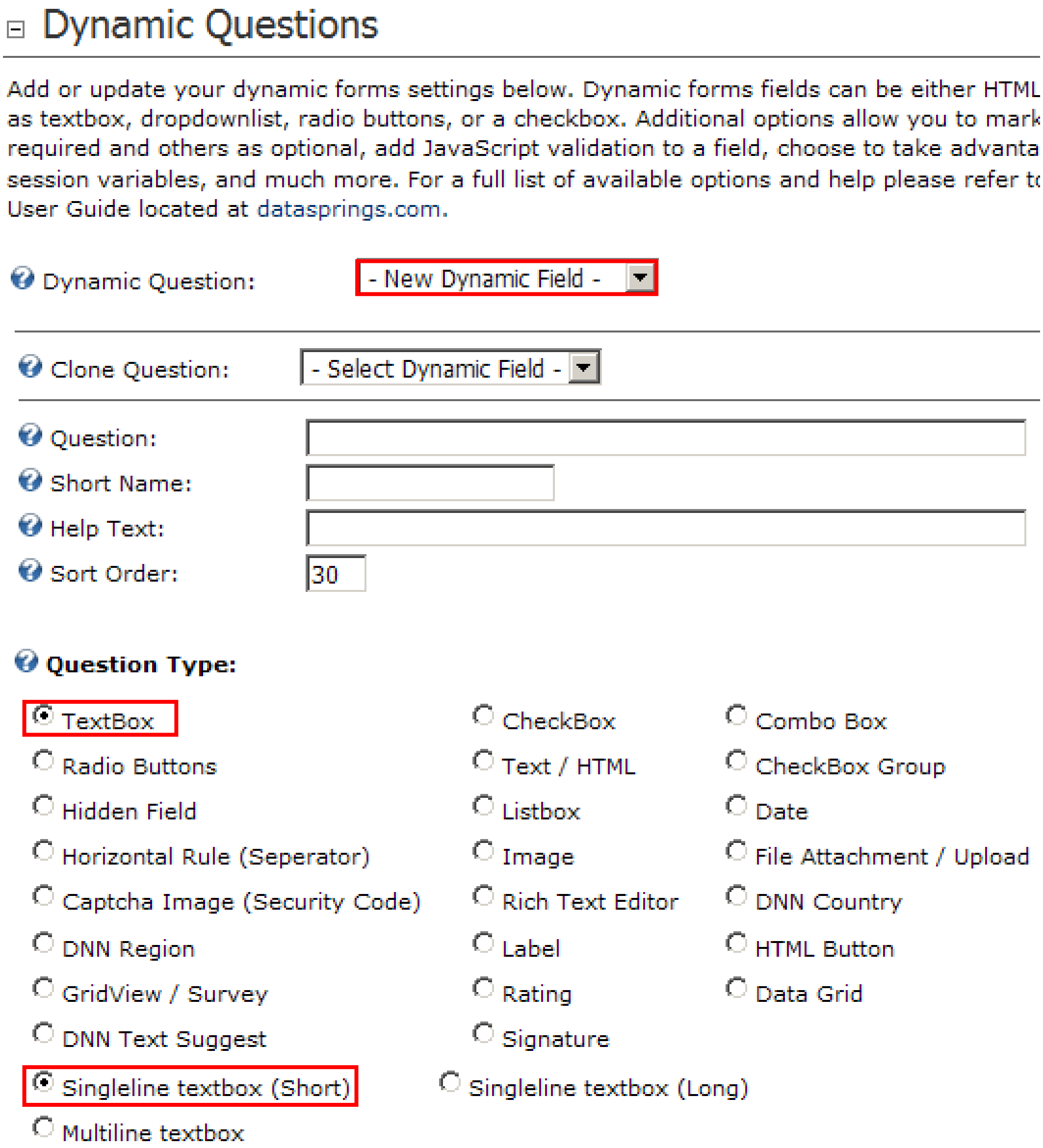


Figure 37: Creating a new question

The following parameters are available inside this screen:

- **Dynamic Question** – input field for entering the question as displayed inside the form on the website (**e.g.** “First name”)
- **Clone Question** – this is a useful feature if you are creating a question which closely resembles the one that has already been created; if that is the case, just choose the question you have already made from this pull down menu and the fields will be populated with its value; you can then proceed onwards to modifying the contents according to the requirements for the new question
- **Question** – enter the question which will be displayed in the form (**e.g.** “First name”)
- **Short Name** – set the short name for the question (**e.g.** “firstname”); this information will be used for creating events triggered based on user responses to this question i.e. customizing email messages sent to the users (see section 7.37)

- **Help Text** – input field for entering help text which will be displayed to the user once he places the mouse over the question mark
- **Sort Order** – set the desired sort order for the question
- **Question Type** – use these options to set the desired type of question i.e. form element
    - TextBox
    - Radio Buttons
    - Hidden Field
    - Horizontal Rule (Separator)
    - Captcha image (Security Code)
    - DNN® Region
    - GridView/Survey
    - DNN Text Suggest
    - Singleline textbox (Short)
    - Singleline textbox (Long)
    - Multiline textbox
    - Checkbox
    - Text/HTML
    - Listbox
    - Image
    - Rich Text Editor
    - Label
    - Rating
    - Combo box
    - CheckBox Group
    - Date
    - File Attachment/ Upload
    - DNN® Country
    - HTML Button
    - Data Grid

After setting the desired parameters, click on the "Update Field" link and the new question will be created.

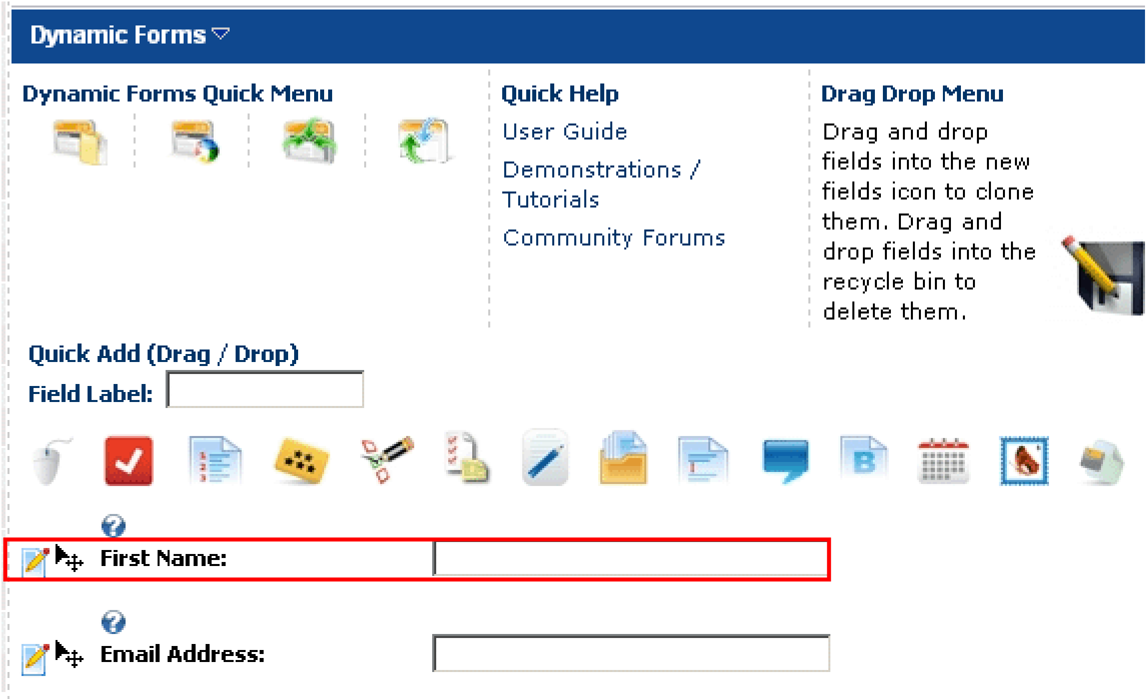


Figure 38: Example of the created question

7.3 Setting the Advanced Field Options

In order to start setting advanced options for the field, click on the "+" symbol next to "Advanced Field Options" label.

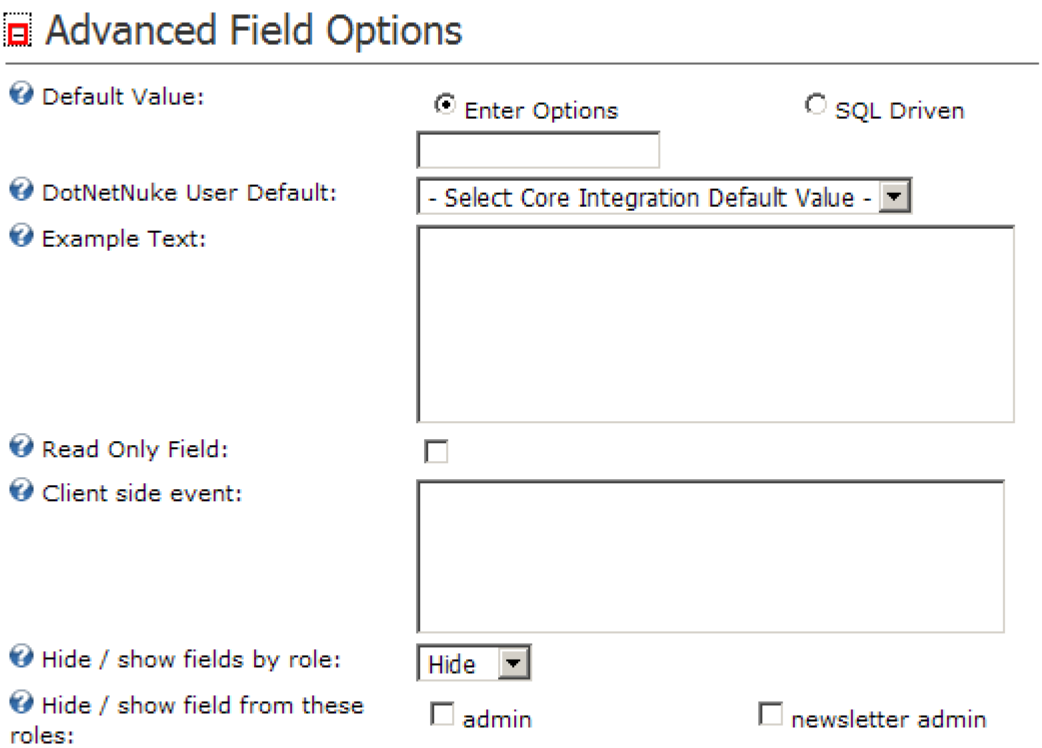


Figure 39: Managing Advanced Field Options (screenshot 1/2)

The following parameters are available inside the first part of the screen:

- **Default Value** – enter default value for this field

- **DotNetNuke® User Default –** this represents the DotNetNuke® user profile fields. You can choose for this field to represent any DotNetNuke® standard or extended profile field and the field will default the profile field attribute.
    - **For example**: if you default the field to ‘First Name’ then when the user navigates to the page their first name will automatically be defaulted into the form field.
- **Example Text** – enter the text you want to be displayed as an example
- **Read Only Field** – If a read only field is checked the user will not be able to change the contents within the field; the fields value would either be set to the default value or a value returned from the query string/session variables.
- **Client Side Event** – client side events are JavaScript / AJAX events which can be executed on focus for a field (see section **7.3.1**)
- **Hide/show fields by role** – use this pull down menu to select if you would like to hide or show the fields by security roles within the form
- **Hide / show field from these roles** – select the roles you wish to hide or show this field for

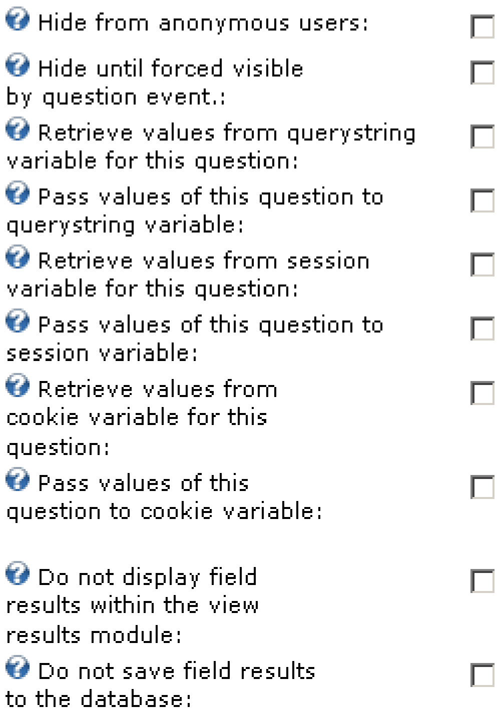


Figure 40: Managing Advanced Field Options (screenshot 2/2)

The following parameters are available inside the second part of the screen

- **Hide from anonymous users** –check this checkbox if you want to hide this question from anonymous users
- **Hide until forced visible by question event** – Check this box if you would like to have the question not display until it should be based on a question event. For example, maybe you would like the question “What version of DotNetNuke® are you running on?” but you don’t want it to be displayed unless they check a previous question called “Do you currently run on DotNetNuke?”. This is the only area of Question Events that is handled outside of the question events area, all other question event details are found in section 7.53.
- **Retrieve values from query string variable for this question** – This will retrieve values passed from the querystring within the URL and insert them into the textbox. The querystring value passed must be the Short Field Name for the field. For example: http://www.datasprings.com/default.aspx?TabID=36&MyFirstName=John&MyLastName=Doe. If the fields First Name/Last Name which was setup with the short field names of myfirstname/mylastname the values would be John Doe immediately after navigating to that page.

- **Pass values of this question to query string variable** – Once the form as been completed Dynamic Forms can optionally pass values to the url the user is directed to. For example http://www.datasprings.com/default.aspx?TabID=36&MyFirstName=John&MyLastName=Doe
- **Retrieve values from session variable for this question** – Similar to Querystring Variables you can accept session variables and default the fields based on the session variable.
- **Pass values of this question to session variable** – similar to passing querystring variables Dynamic Forms can also set session variables. These session variables can later be used by other modules on the site for various purposes
- **Retrieve values from cookie variable for this question** – select this option to retrieve values from the cookie variable (if available) for this field. Cookie values passed must represent the question shortfieldname
- **Pass values of this question to cookie variable** – select this option to pass the values of this question to the cookie variable
- **Do not display field results within the view results module** – select this option if you want to hide the results of this field from the view results section of Dynamic Forms (**note**: this setting will skip this field when viewing form results or exporting results to Excel)
- **Do not save field results to the database** – select this option if you do not wish to save the result of the field to the database; often you might want to process data but don't want the data ever stored within the database DynamicForms_QuestionResponse table.
    - **For example:** if you are processing credit cards and collecting sensitive data such as the credit card number; you might not want to save that data to the database once it has been processed.
- **Encrypt / decrypt field results** – select this option if you want to encrypt the form results then decrypt them within the view results section

After setting the desired parameters, click on the "Update Field" button to save the changes.

7.3.1 Setting up client side events

In order to setup a client side event for the desired dynamic question you need to click on the "+" symbol next to the "Advanced Field Options".

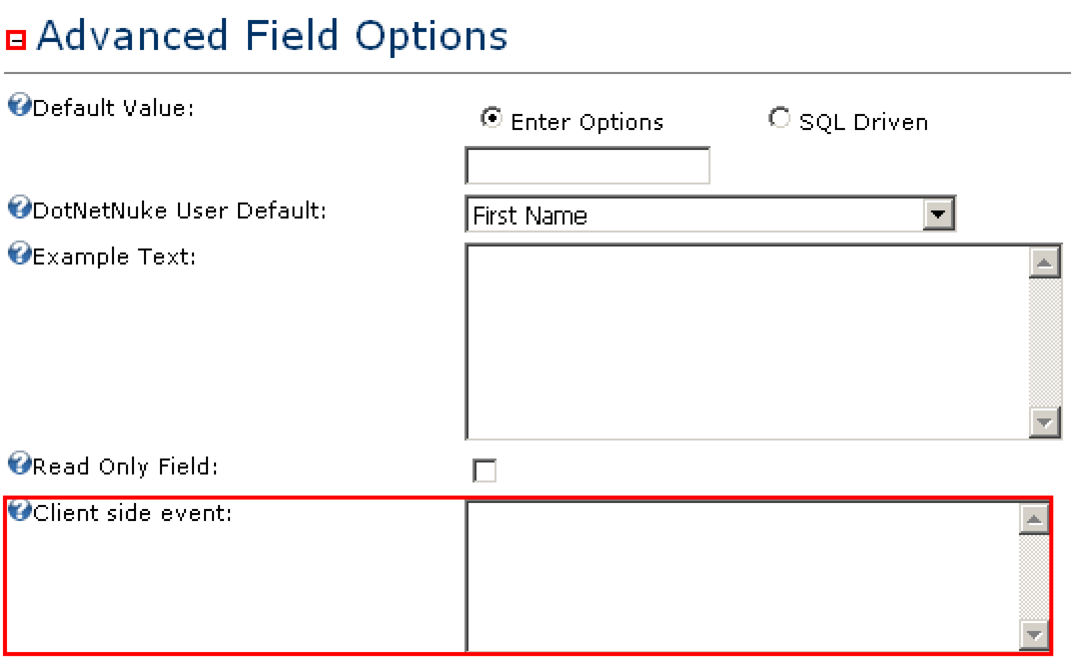


Figure 41: Setting up client side events

Enter the desired AJAX or JS code which will be applied to this dynamic question inside the “Client side event” text area.

Note: for further information and examples, see section 7.40.

7.4 Setting up Question Look and Feel

In order to adjust the layout of the desired question click on the “+” symbol next to the “Question Look/ Feel” label

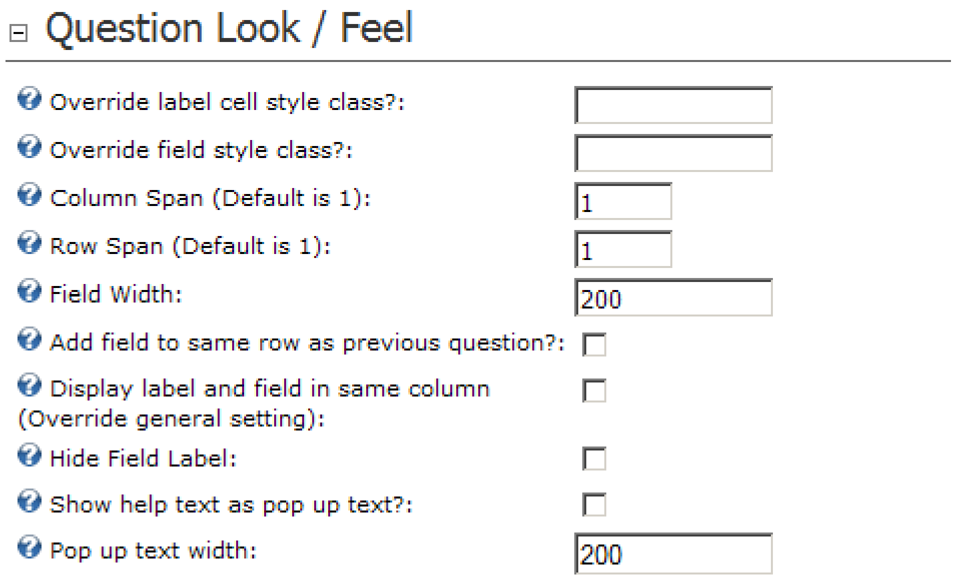


Figure 42: Setting up Question Look and Feel

The following parameters are available inside this screen:

- **Override label cell style class?** - enter a style class if you would like to override the standard class
- **Override field style class? -** enter a style class if you would like to override the standard class
- **Column Span (Default is 1)** – use this input field to specify the desired value for he column span. This feature can be useful if you are using larger fields such as Rich Text Editors or Multi Line Textbox fields and you are also using multiple columns of fields (adding fields to the same row as previous question) within your form, and the form width is expanding more than you would like between columns.
- **Row Span (Default is 1)** - use this input field to specify the desired value for the row span. Similar to column span, this feature can be useful if you are using larger fields such as Rich Text Editors or Multi Line Textbox fields and you are also using multiple columns of fields (adding fields to the same row as previous question) within your form, and the form width is expanding more than you would like between rows within the form.
- **Field Width** - select the desired length for this field
- **Add field to same row as previous question?** – select this checkbox if you wish to you like to add fields to the same row as the previous question
- **Display label and field in same column (Override general setting)** - select if you would like to override the default setting and force the field to be on the same column as the question label
- **Hide Field Label** – select this checkbox in case you wish to hide the field label
- **Show help text as pop up text** – select this checkbox to show the help text within a pop up
    - **Note:**

- You can setup settings for the pop up (bubble up) help text within the stylesheet area (see section 7.38). The default settings are:

```
#hintbox{ /*CSS for pop up help box */
position:absolute;
top: 0;
background-color: lightyellow;
width: 150px; /*Default width of pop up help.*/
padding: 3px;
border:1px solid black;
font:normal 11px Verdana;
line-height:18px;
z-index:100;
border-right: 3px solid black;
border-bottom: 3px solid black;
visibility: hidden;
}
```

- see section 7.4.1 for examples of the pop up help text

* **Pop up text width** – use this field to specify the desired width for the pop up text (value should be entered in pixels - 7.4.1)

7.4.1 Example of the help pop up text

This section give an example of the help pop up text.

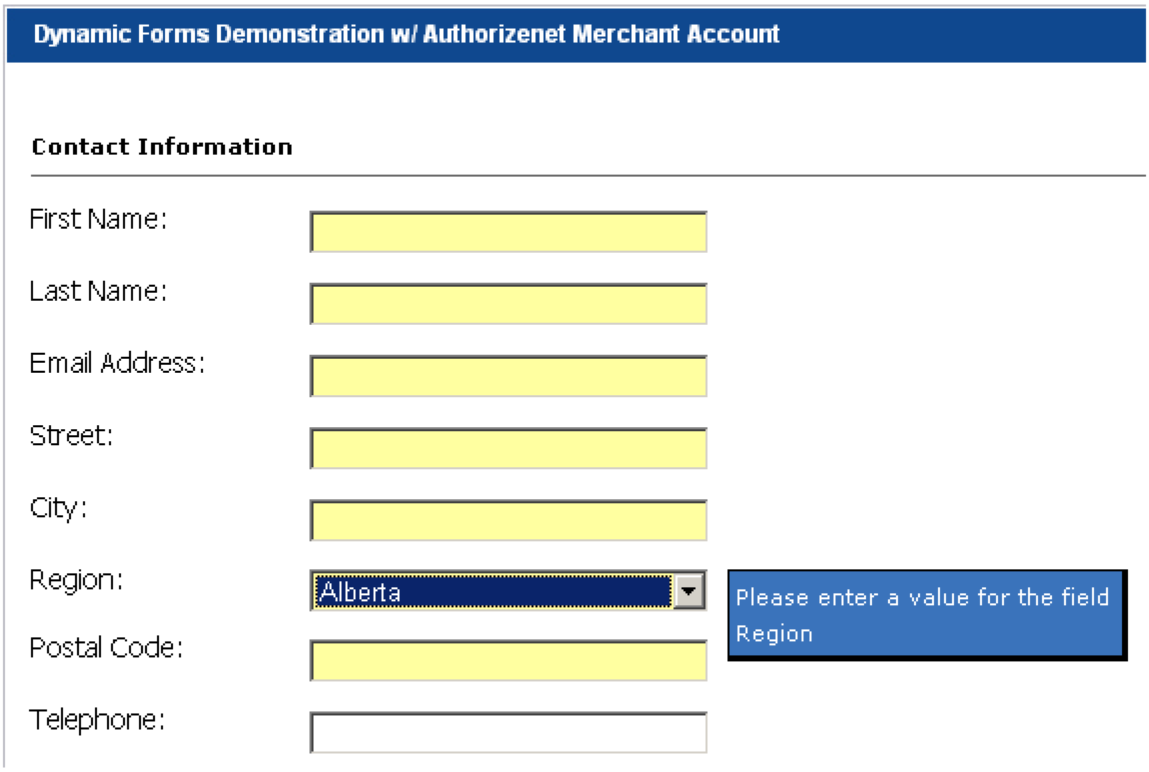


Figure 43: Help pop up example 2

7.5 Setting up Question Header Footer

In order to setup the question header or footer click on the "+" symbol next to the "Question Header/Footer" label inside the screen for creating a new question.

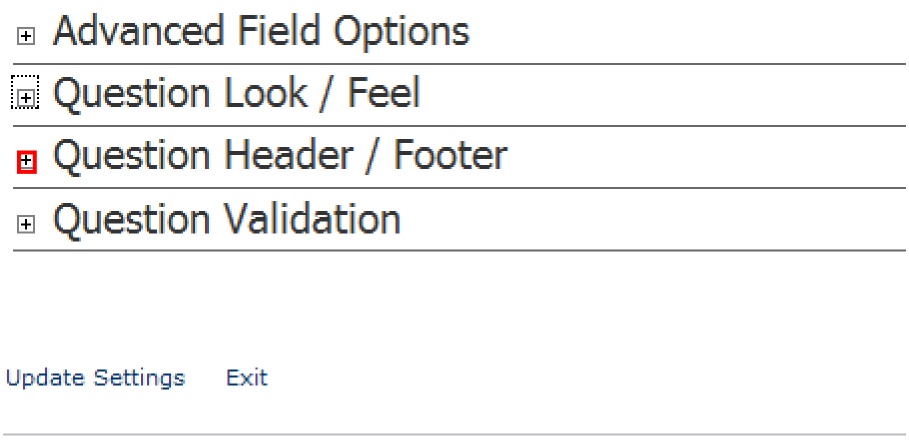


Figure 44: Choosing option "Question Header/Footer

The following screen will be displayed.

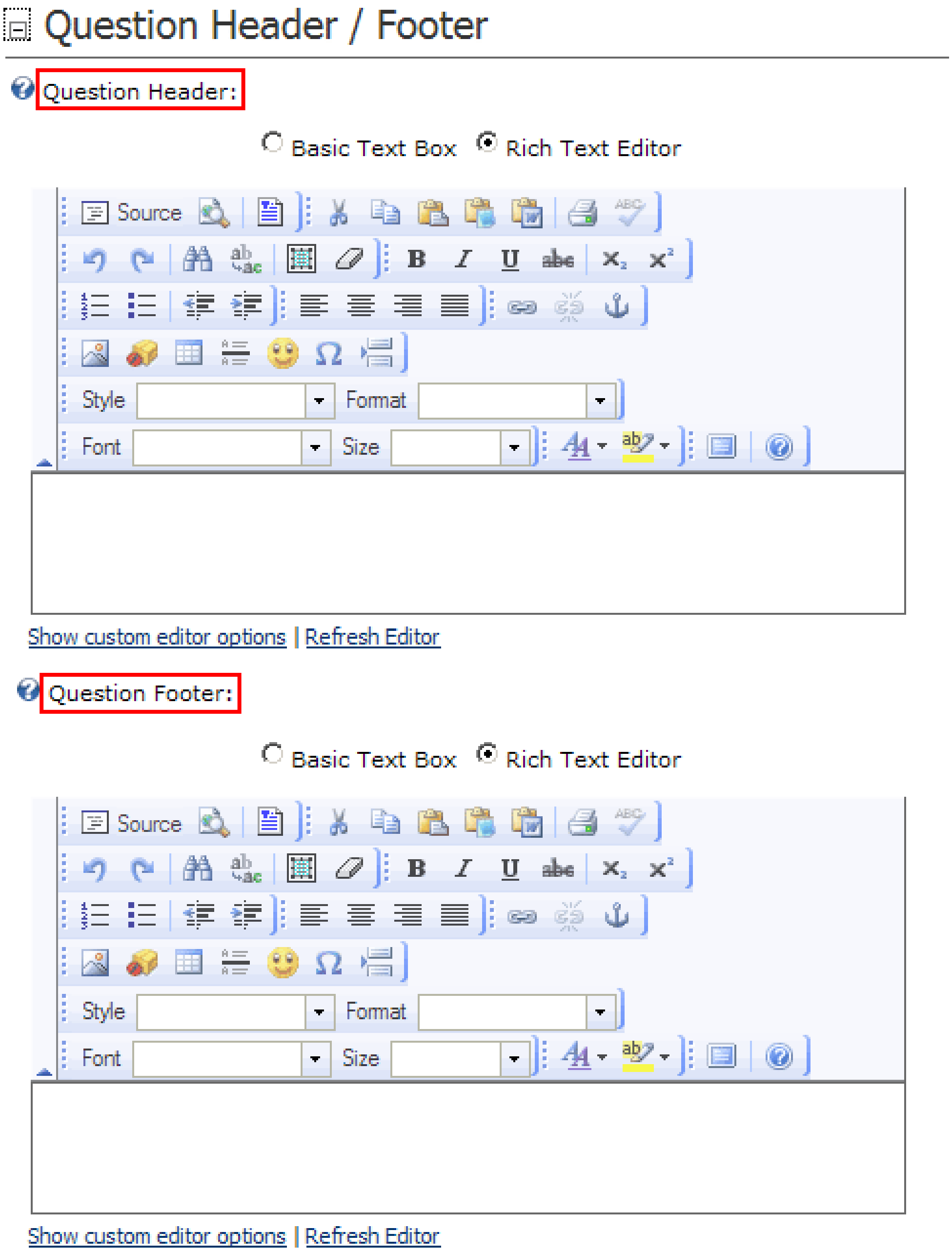


Figure 45: Setting up question header or footer

The following parameters are available inside this screen:

- **Header** – text area for defining the question header
- **Footer** – text area for defining the question footer

7.6 Managing Question Validation

In order to start managing the question validation options click on the “+” symbol next to the “Question Validation” label, inside the screen for creating a new question.

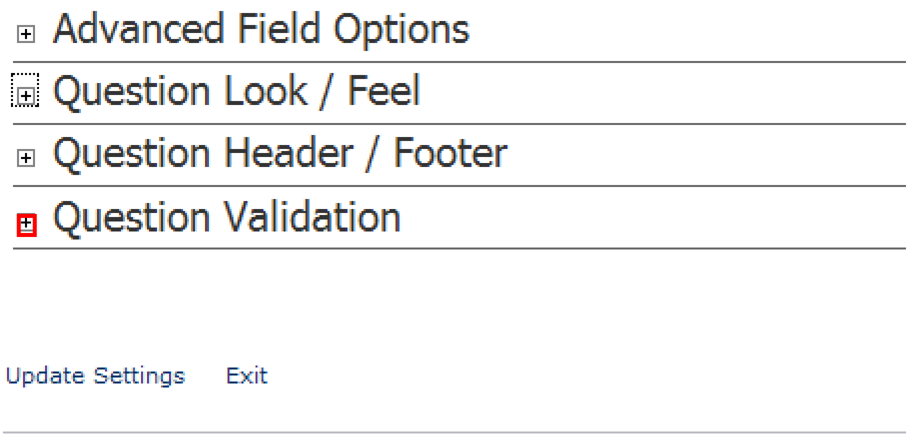


Figure 46: Choosing option "Question Validation"

The following screen will be displayed.

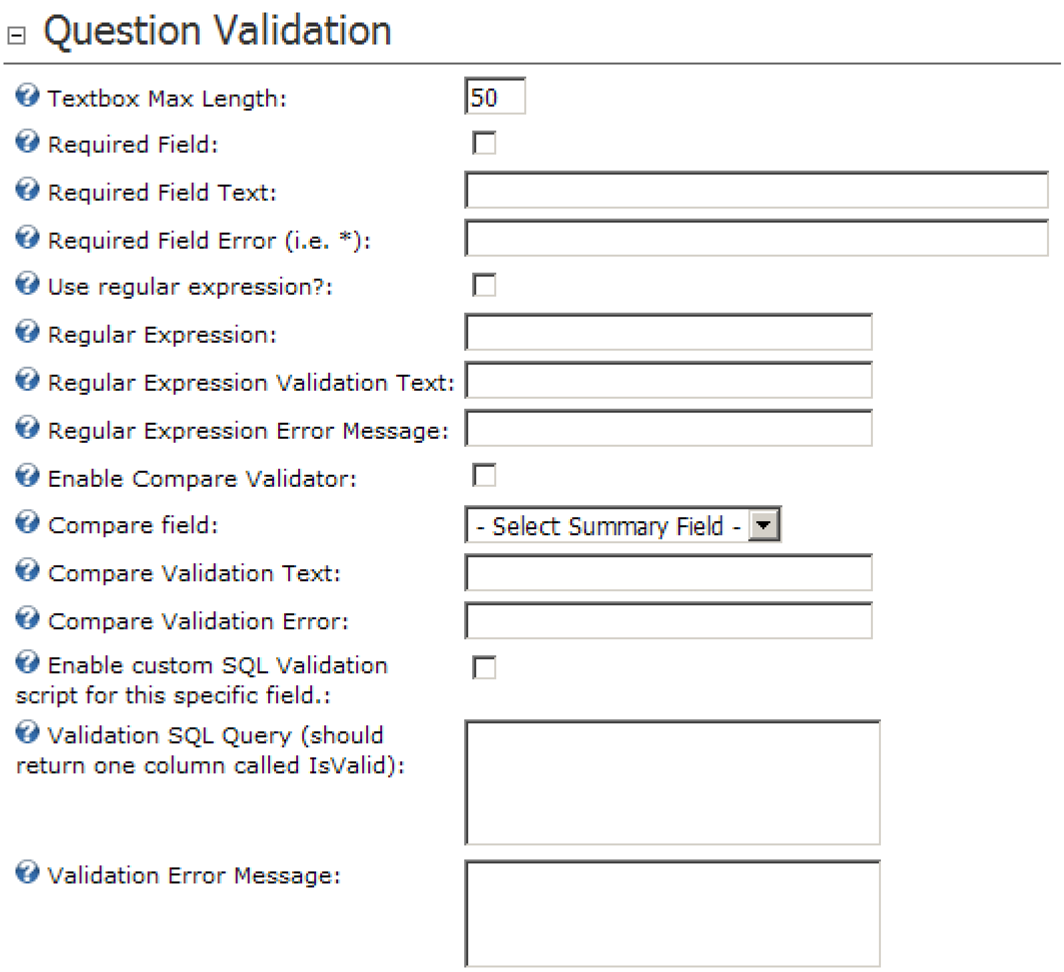


Figure 47: Managing question validation

The following parameters are available inside this screen:

- **Textbox Max Length** – define the maximum field length (in the number of characters)

- **Required Field** – select this checkbox if the question is supposed to be a required field
- **Required Field Text** – this is the text that the user will be presented with if they do not fill in the required field. This text will either be displayed immediately if client side validation is turned on under general settings, or after the submit button is clicked (if server side validation is turned on under general settings)
- **Required Field Error (i.e. *)** – field for entering the text which will be displayed in case the user forgot to supply information for this field because it is required
- **Use Regular Expression** – select if you would like to use regular expression validation for this field. Regular expression validations can be found on various web sites (try any search on the web for example regular expression validators). Regular expressions can force the user to enter data into a correct format. For example, maybe you want to force the user to enter in 5 numbers exactly, or maybe xxx-xxx-xxx etc… There are literally hundreds of regular expressions you can select from, if you need help finding a specific expression please post a thread to the Data Springs Product Forum and a support representative will be happy to respond.
- **Regular Expression** - enter in the expression for this validation, there are a few examples below:
  - Find the whole word 'yes' within the textbox. You could force the user to type in yes within the textbox as a validation constraint before they submit the form.
    - Expression: "\yes\b"
  - Validate phone number is entered as ###-###-####
    - Expression: "\(?\d{3}[-.]\)? ?\d{3}[-.]\d{4}"
  - Social Security Number
    - Expression: "\d{3}-\d{2}-\d{4}"
  - Email Address Validation
    - Expression: "^[_a-zA-Z0-9-]+(\.[_a-zA-Z0-9-]+)*@[a-zA-Z0-9-]+(\.[a-zA-Z0-9-]+)*\.(([0-9]{1,3})|([a-zA-Z]{2,3})|(aero|coop|info|museum|name))$"
- **Regular Expression Validation Text** – the field for entering the regular expression validation text for the field; please note that this is not the regular expression itself, but the error text that the user will receive in case the validation was false
- **Regular Expression Error Message** – this is the message the user will receive if they do not enter the correct formatted expression into the field. For example ('You must enter a valid U.S. phone number'). The validation text and validation error message can either be the same, or you could have the error message next to the field as an * and the validation text would only appear in the error summary at the top or bottom of the page.
- **Enable Compare Validator** – select this checkbox in order to enable the compare validator option
- **Compare Field** – use this pull down menu to choose the field which will be used for comparison
- **Compare Validation Text** – field for entering the compare validation text
- **Compare Validation Error** - field for entering the compare validation error text
- **Enable custom SQL Validation script for this specific field** – select this option if you would like to enable a custom SQL Validation Script for this specific vield; validation via SQL should ALWAYS use stored procedures to prevent SQL Injection when referencing fields
- **Validation SQL Query (should return one column called IsValid)** - enter an SQL query to perform custom validation for this field.; the SQL query should return one column called IsValid; If the column returns 0 or False then the validation error message will be thrown, anything else and the user can continue; validation via SQL should ALWAYS use stored procedures to prevent SQL Injection when referencing fields

- **Validation Error Message** - enter a message the user will receive if the validation is performed via custom SQL Validation

After setting the desired parameters, click on the "Update Field" link to save the changes.

7.7 Creating a Textbox

The purpose of the textbox element is to allow the user to input text information to be used by the program. In order to start creating textbox element, choose option “TextBox” once inside the screen for creating a dynamic question.

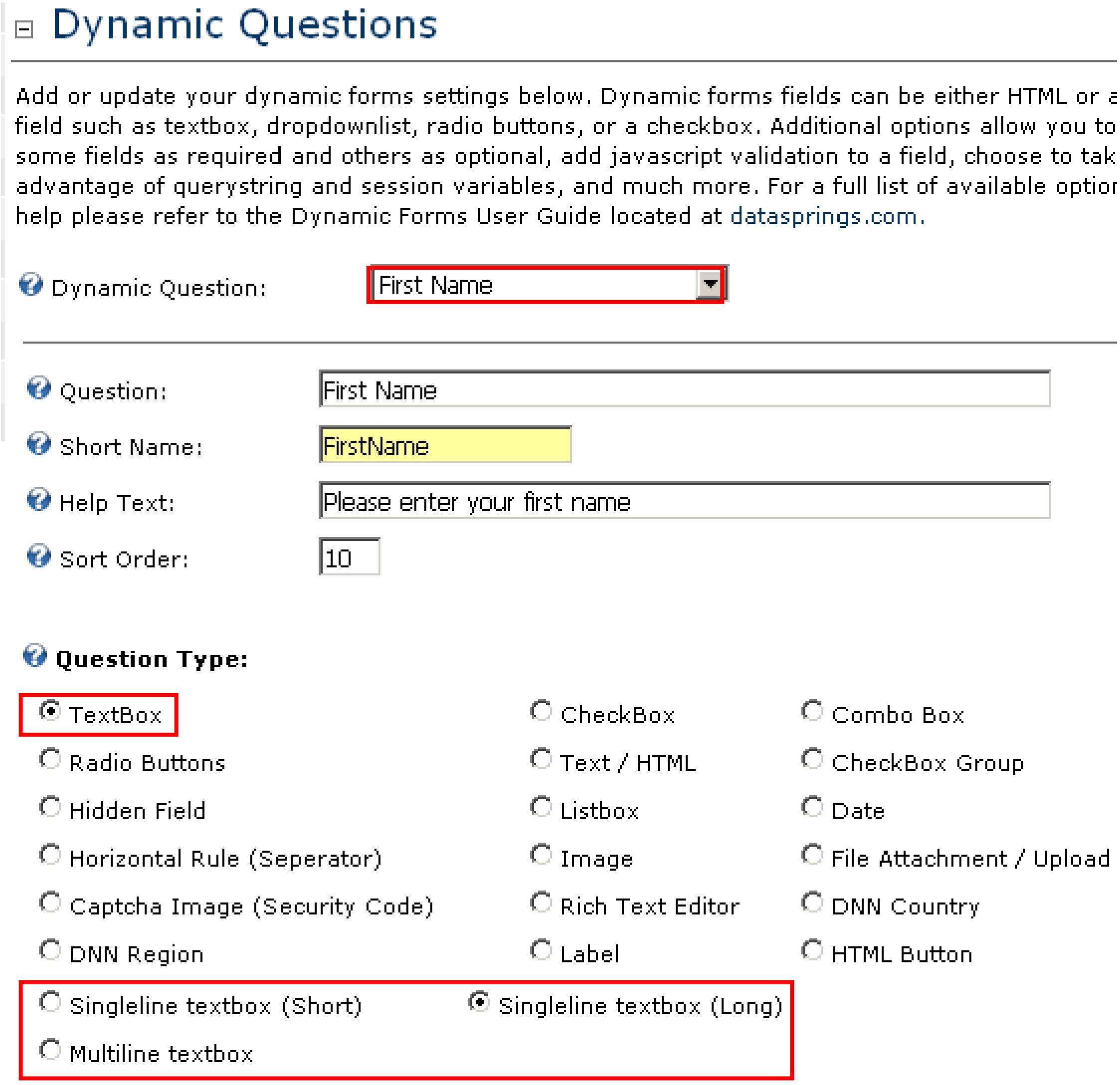


Figure 48: Creating a textbox element

The following parameters are available inside this screen:

- **Question** – input field for setting the question that will be displayed in front of the textbox (**e.g.** “First Name”)
- Short Name
- **Question Type** – choose option “TextBox” and then you have the following subtypes
  - **Singleline textbox (Short)** – see section 7.8
  - **Singleline textbox (Long)** – see section 7.9
  - **Multiline textbox** – see section 7.10

7.8 Creating a Singleline textbox (Short)

In order to create a short singleline textbox, choose option “Singleline textbox (Short)”. This is a shorter textbox that can be used for user’s first name, last name, username, etc.

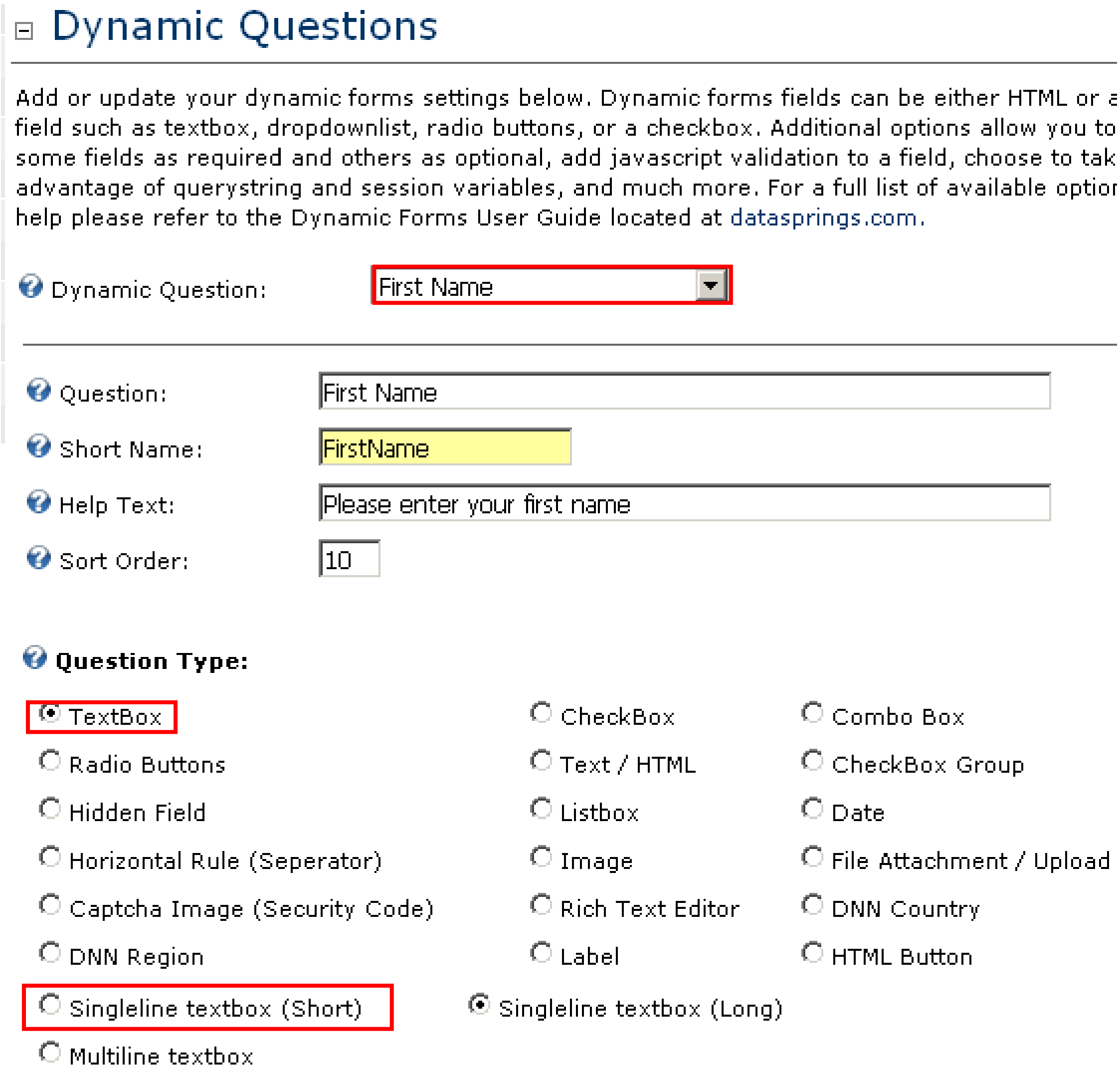


Figure 49: Creating a singline textbox (Short)

After setting the desired parameters, click on the “Update Field” in order to complete the procedure of creating a Singleline Textbox (Short).

The following screenshot demonstrates the shorter singleline textbox as seen by the end user.

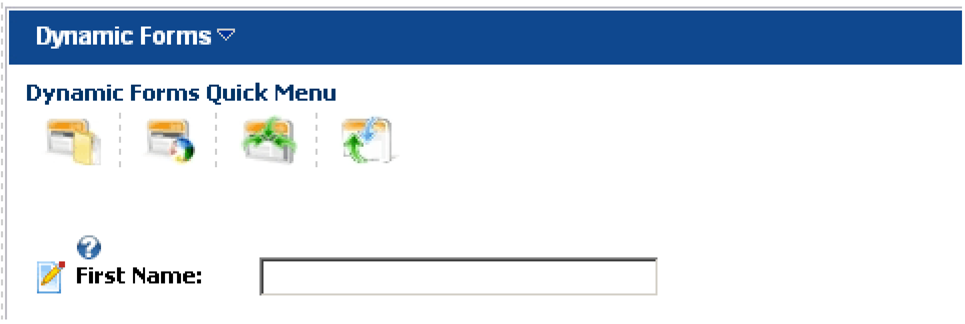


Figure 50: Example of the Singleline Textbox (Short)

7.9 Creating a Singleline textbox (Long)

In order to create a long singleline textbox, choose option “Singleline textbox (Long). This is a longer textbox that can be used for acquiring longer text from the user, such as address.

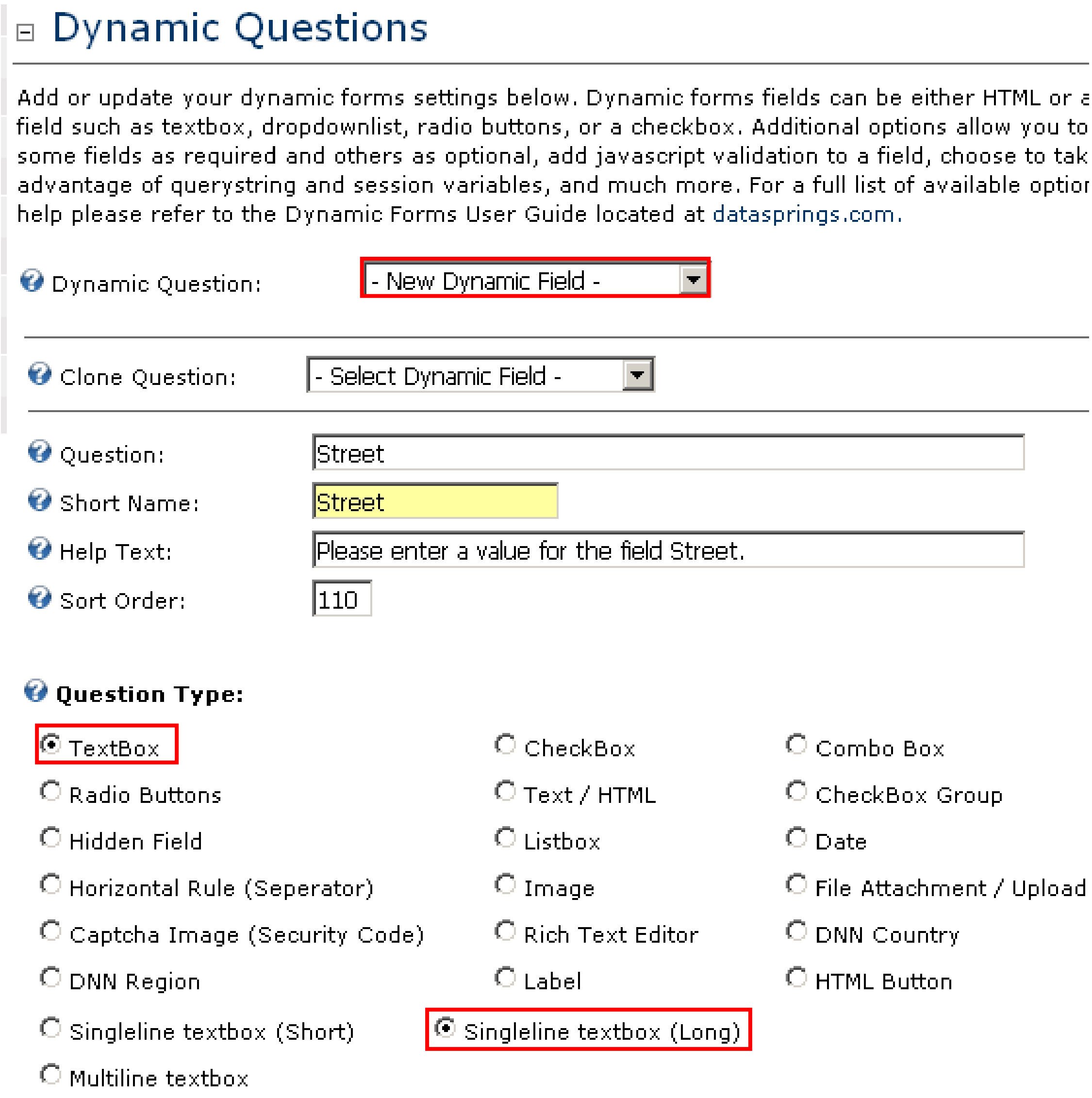


Figure 51: Creating a singline textbox (Long)

After setting the desired parameters, click on the “Update Field” in order to complete the procedure of creating a Singleline Textbox (Long). The following screenshot demonstrates the shorter singleline textbox as seen by the end user.

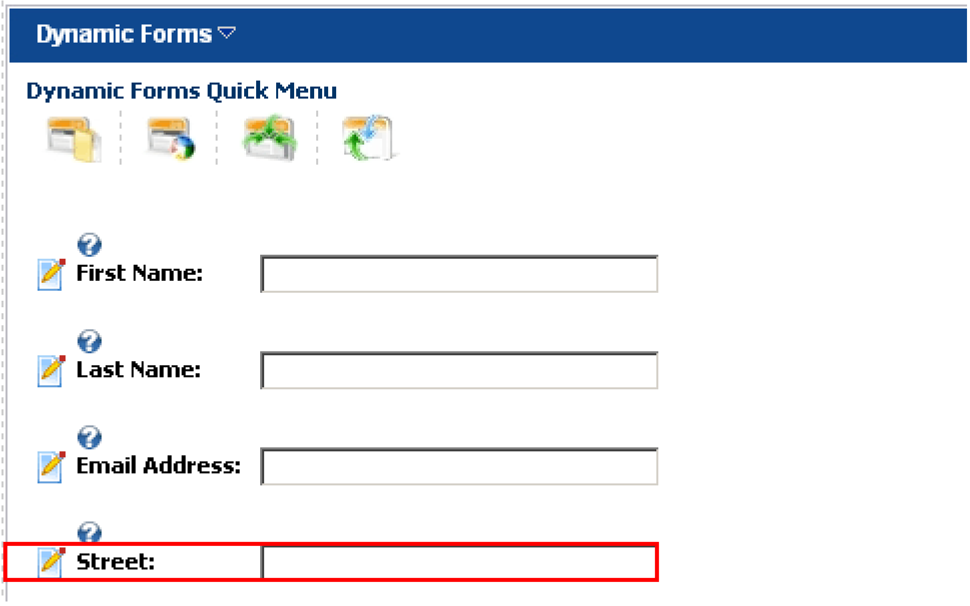


Figure 52: Example of the Singleline Textbox (Long)

7.10 Creating a Multiline textbox

This element should be used in case you want a longer feedback from users, for example, give them the opportunity to post any additional comments. In order to create a multiline textbox, choose option “Multiline textbox”.

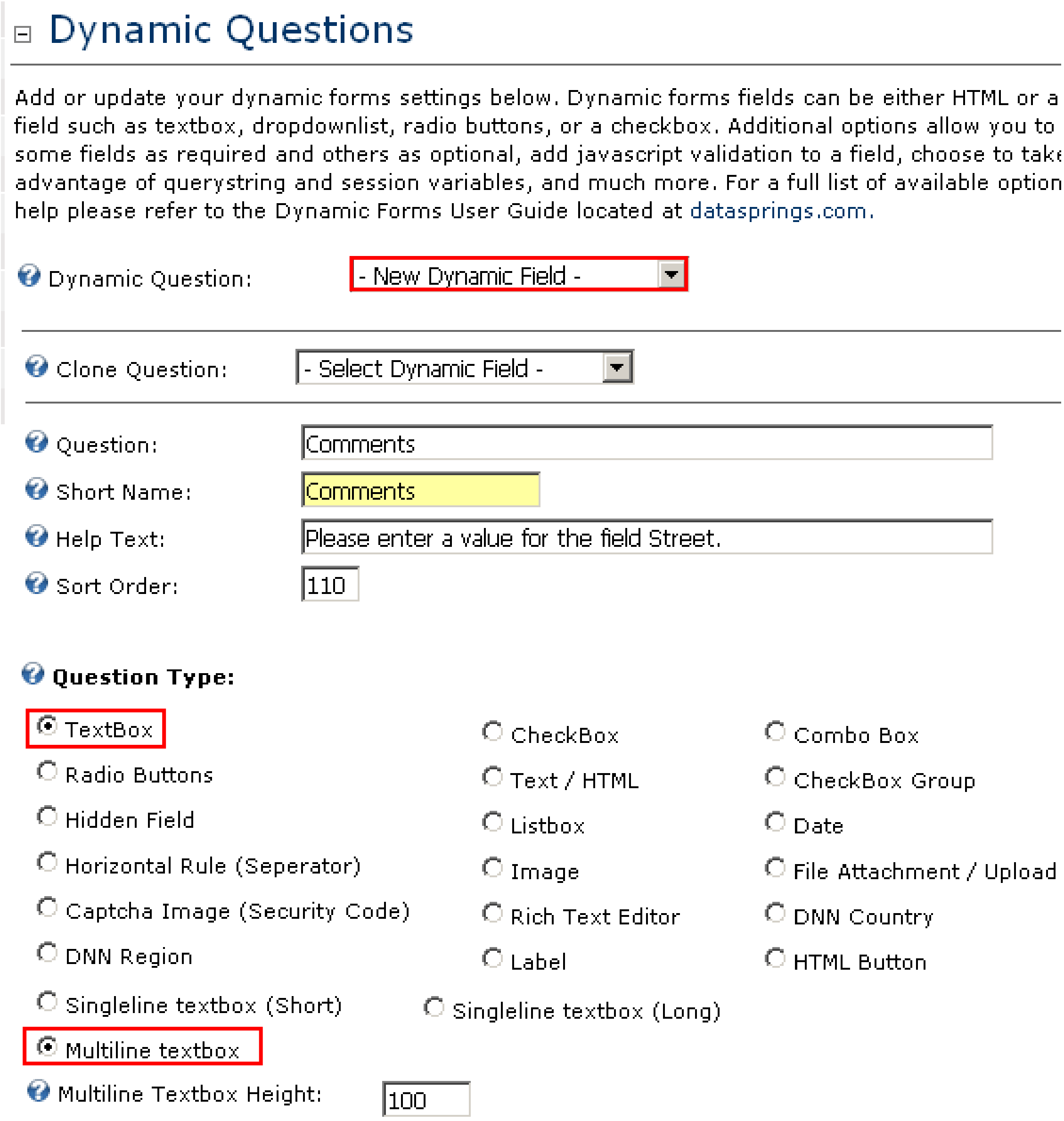


Figure 53: Creating a Multiline textbox

After setting the desired parameters, click on the “Update Field” link in order to complete the procedure of creating a multiline textbox. The following screenshot demonstrates the multiline textbox as seen by the end user.

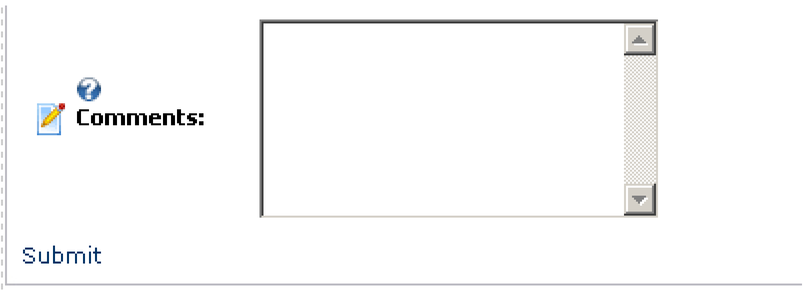


Figure 54: Example of the Multiline textbox

7.11 Creating a Radio Button (Options)

Radio button element is used if you want to have a fixed set of choices where only one of the options in the set can be selected at a time.

In order to start creating radio buttons, choose option “Radio Button” once inside the screen for creating a dynamic question.

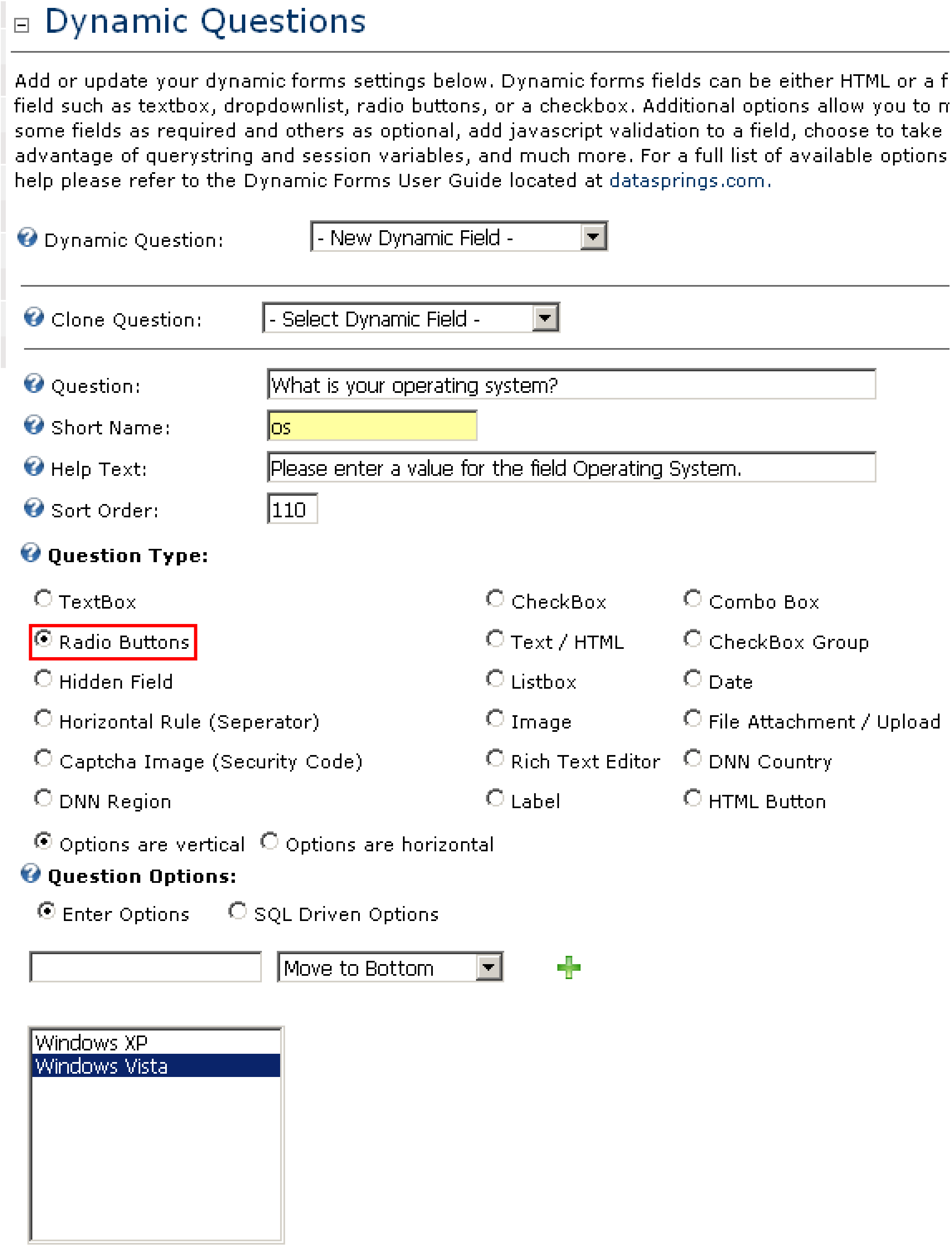


Figure 55: Creating a radio button (step 1/3)

Once you choose the “Radio Buttons” option, the screen will be refreshed containing options for setting the desired orientation for the options (vertical or horizontal) as well as defining the options for radio buttons.

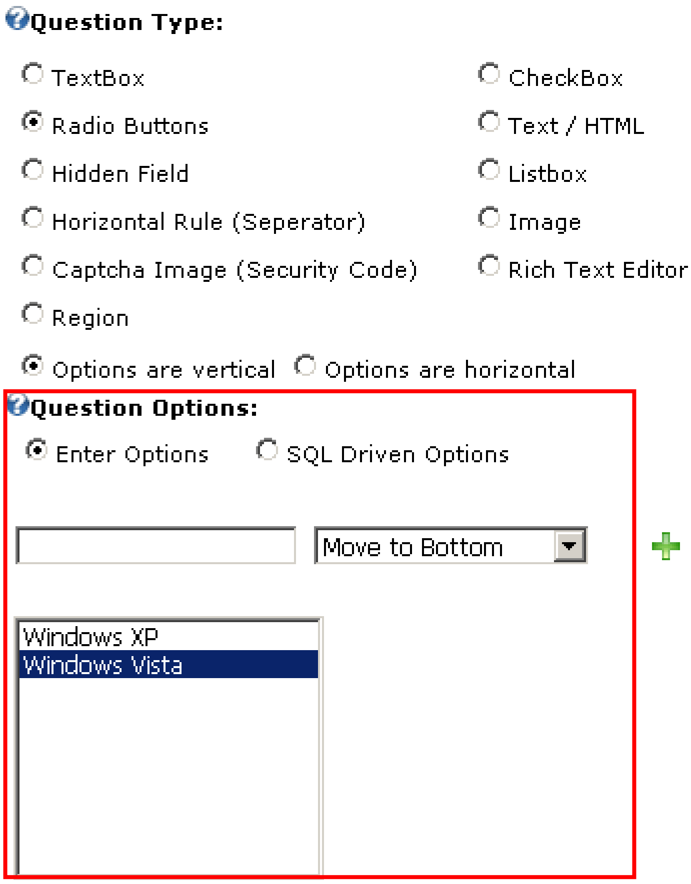


Figure 56: Creating a radio button (step 2/3)

The procedure for defining the options is discussed below.

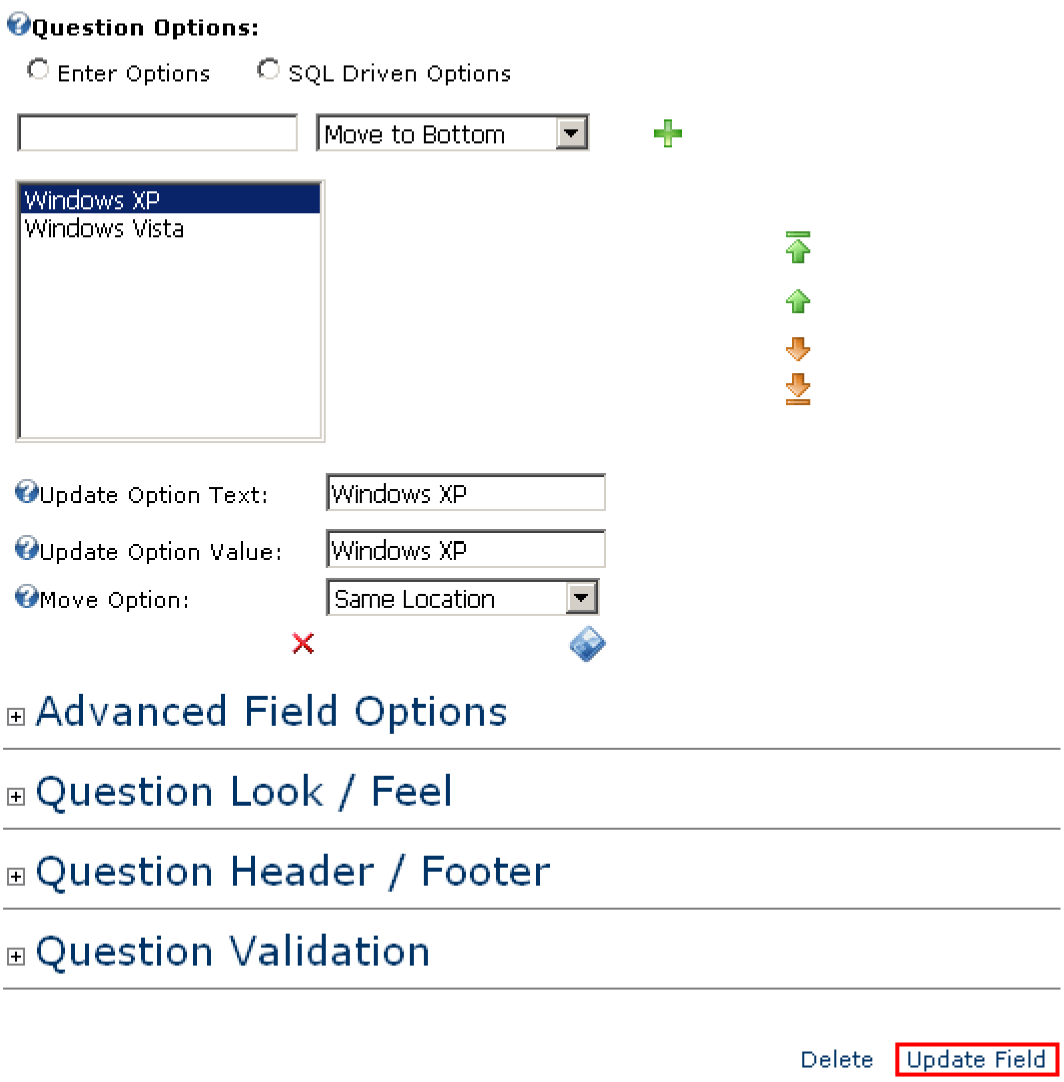


Figure 57: Creating a radio button (step 3/3)

The following parameters are available inside this screen:

- **Enter Options** – choose this option if you want to enter the options manually
- **SQL Driven Options** – use this option in order to automatically insert options from the database, if you already have them defined so that you wouldn't have to do it manually (see section 7.18.1)
- - button for adding the option once you've entered the option name in the input field
- - button for deleting the option; choose the desired option in the list of options and click on this button in order to delete it
- - buttons for setting the desired sort order for the options; select the desired option and click on the up or down arrow to move the option either to the **top** or to the **bottom**
- - buttons for setting the desired sort order for the options; select the desired option and click on the up or down arrow to move the option **up or down one position**
- **Update Option Text** – use this field to update the option text
- **Update Option Value** – use this field to update the option value
- **Move option** – pull down menu for changing the option location

7.11.1 Example for using radio buttons

One example of using the radio buttons for registration form can be asking the user to provide information about his operating system.

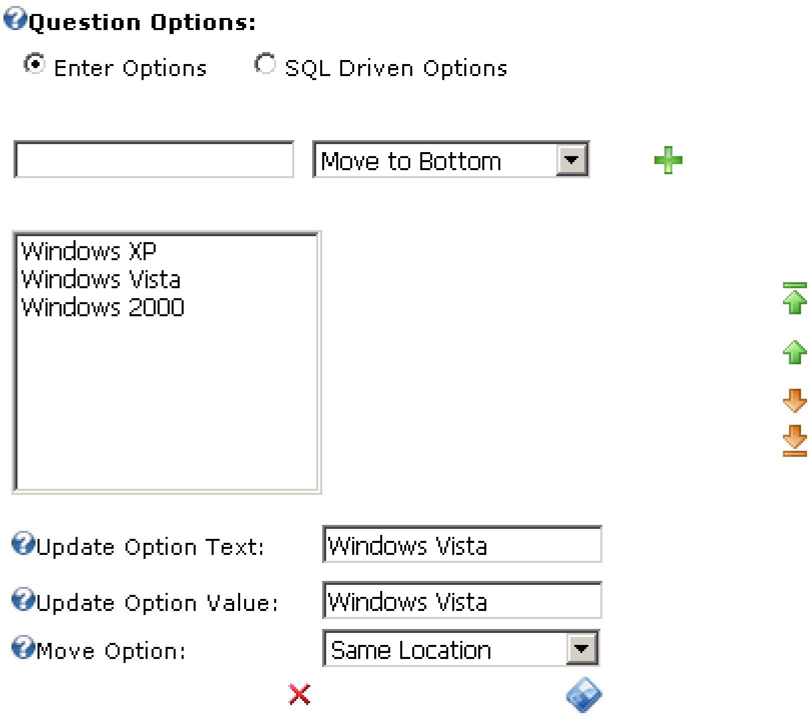


Figure 58: Example of using radio buttons

The parameters would be defined like this:

- **Question** – enter the “Operating System” as a label for the radio buttons set, i.e. informing the user of the requested information
- **Question type** – choose “Radio button”
- **Question Options** – choose “Enter Options”

After setting these parameters, enter the name of the first option inside the input field **e.g.** “Windows XP” and click on this icon. The option will be added to the list of options. You can repeat this procedure for as many options as you like:

- Windows 2000
- Windows 98, etc

Note: use the up and down arrows to set the desired sort order, to update the option and the icon to delete the option.

After setting the desired options and radio buttons, you can click on the “+” symbol next to the “Advanced Field Options” in order to set the default value i.e. the radio button that will be selected by default in your registration form.

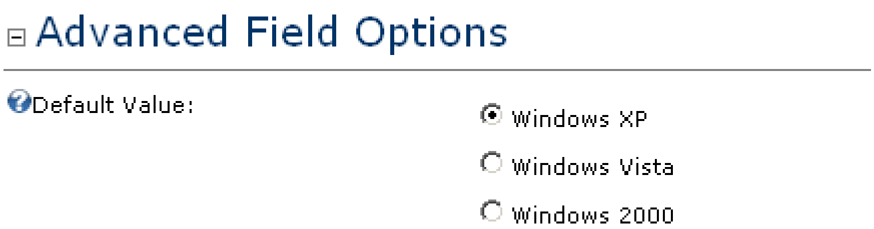


Figure 59: Setting radio button advanced options

Select the radio button you want your users to see as selected by default and click on the “Update Field’ in order to complete the procedure of creating a set of radio buttons.

The following screenshot demonstrates the created radio buttons set as seen by the user on your website.

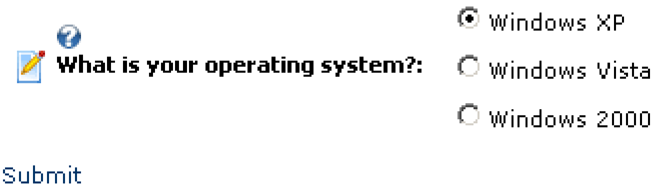


Figure 60: Example of the radio buttons set as seen by the end user

7.12 Creating a Horizontal Rule (separator)

The “Horizontal Rule (Separator)” is used when you wish to divide logical sections within your form. For example if you want to divide personal information (first name, last name, etc.) from information related to user account (username, password, etc.).

In order to create a horizontal rule, choose option “New Dynamic Field” and select option “Horizontal Rule (Separator)”.

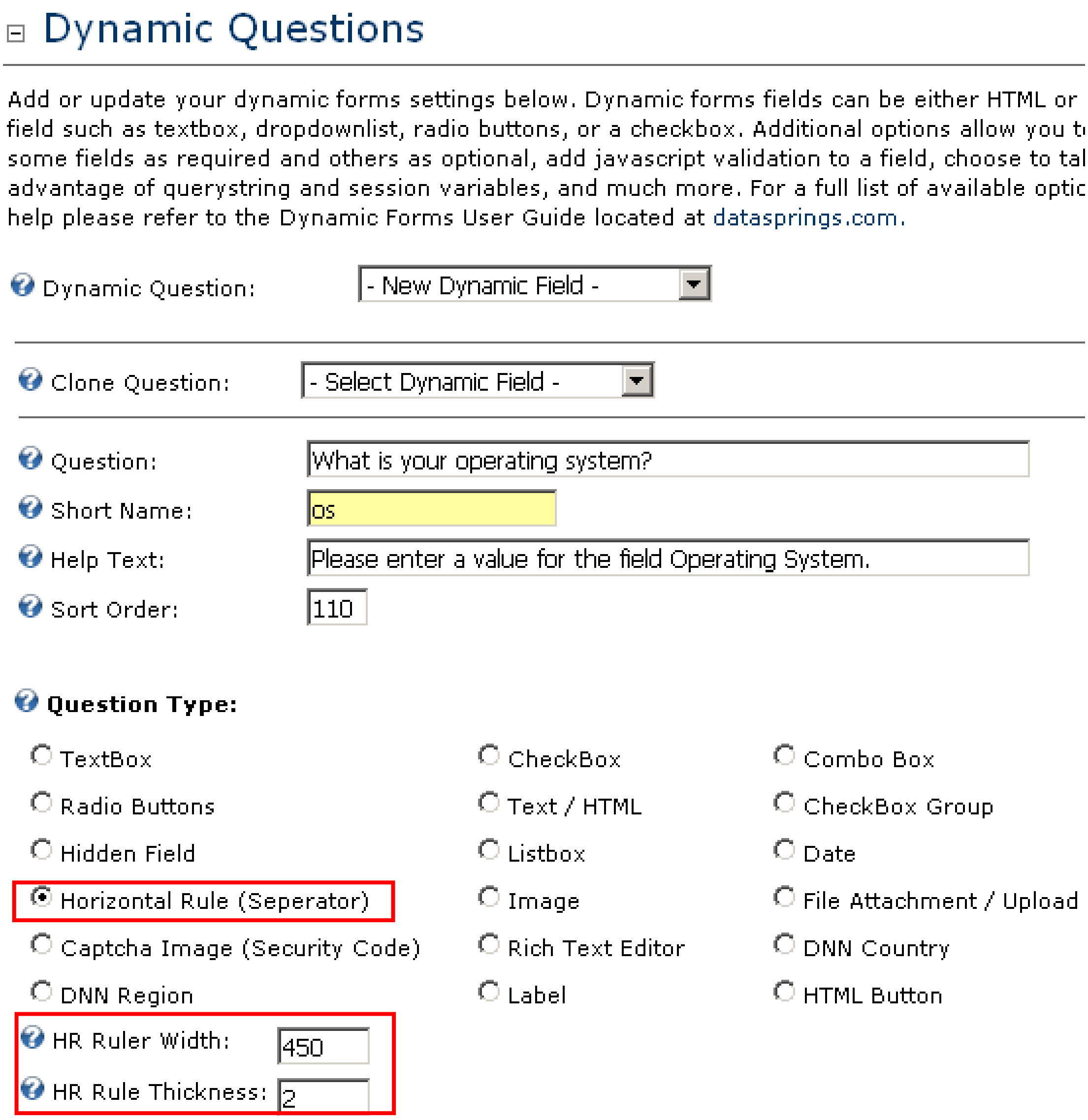


Figure 61: Creating a horizontal rule

The screen will be refreshed and the following horizontal rule parameters will be available:

- **HR Ruler Width** – field for setting the HR ruler width in pixels (**e.g.** 450)
- **HR Rule Thickness** - field for setting the HR ruler thickness in pixels (e.g. 2)

After setting the desired values, click on the "Update Field" link and the horizontal rule will be created.

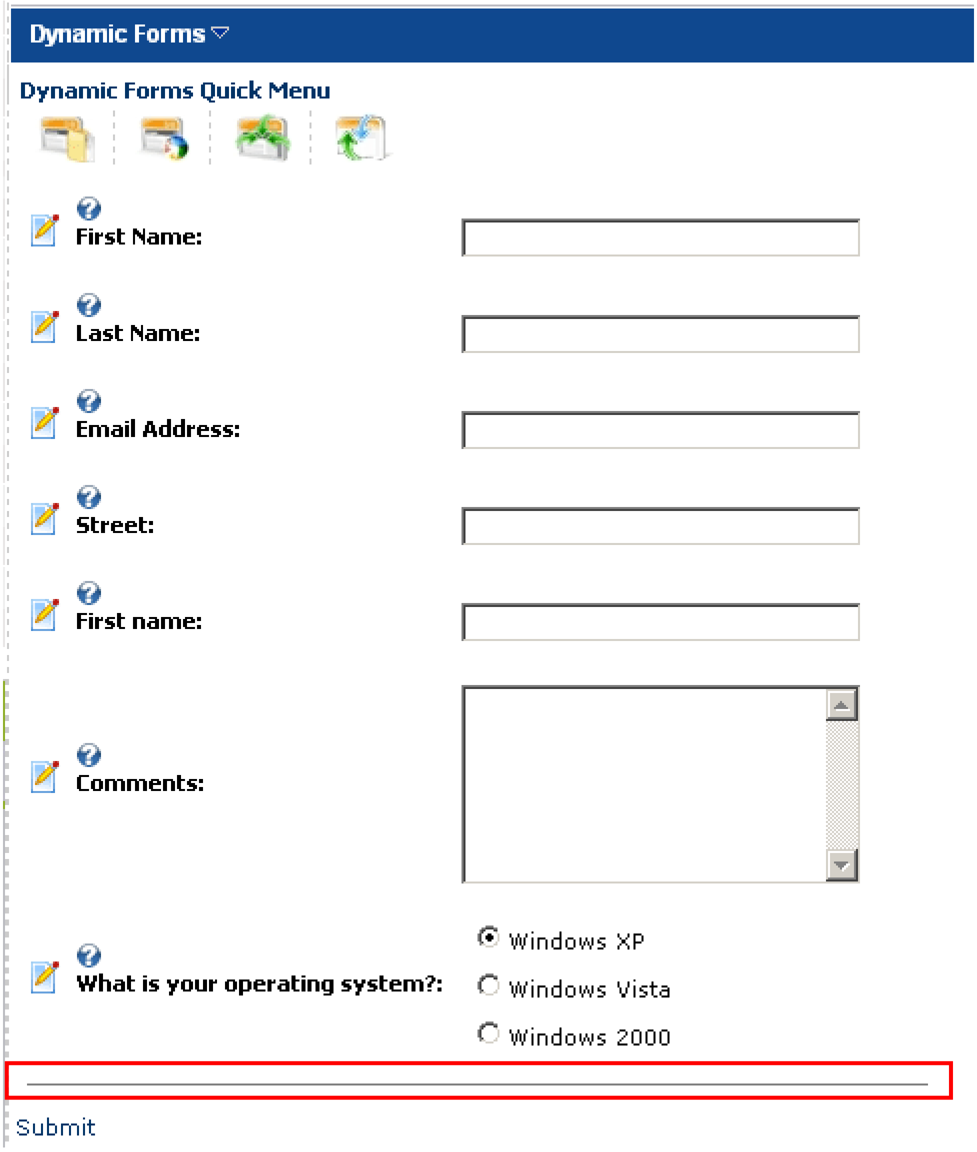


Figure 62: Example of the created horizontal rule

7.13 Captcha Image (Security Code)

The “Captcha Image” element is used as an additional measure of security when making sure the person replying to the questions in the form is human.

This is done by forcing the users to repeat the characters displayed in the automatically created random set of characters (**note**: see http://en.wikipedia.org/wiki/Captcha for further information).

In order to create a “Captcha Image” element, choose the “Captcha Image (Security Code)” option.

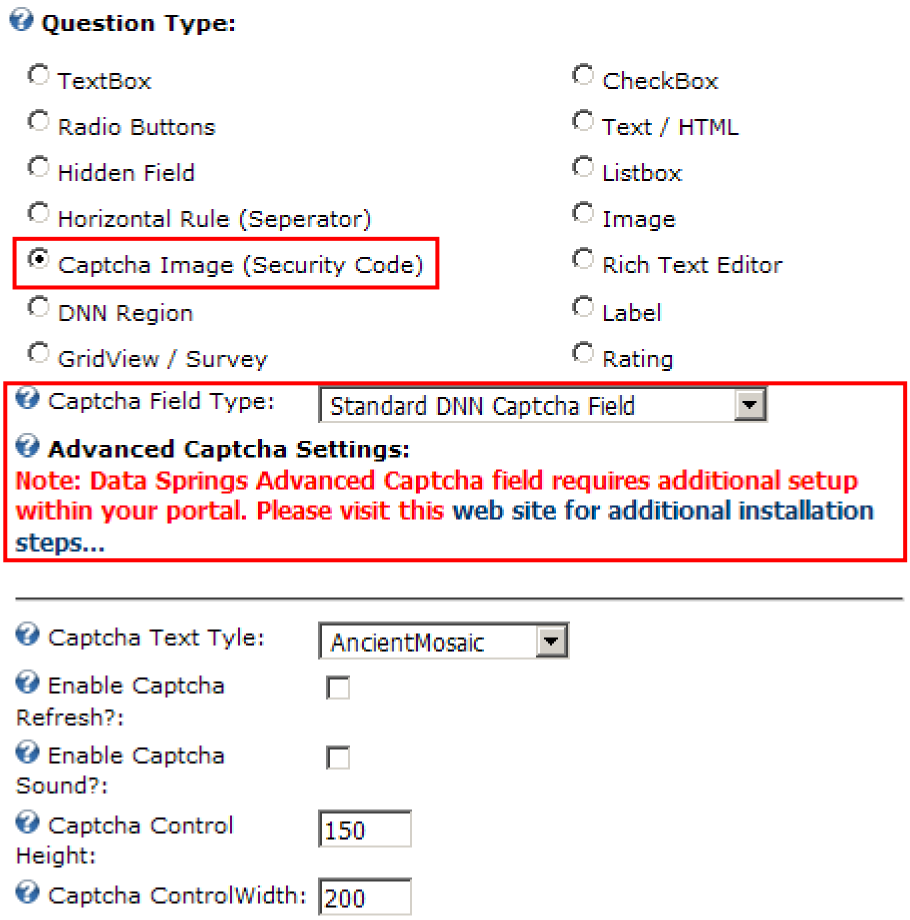


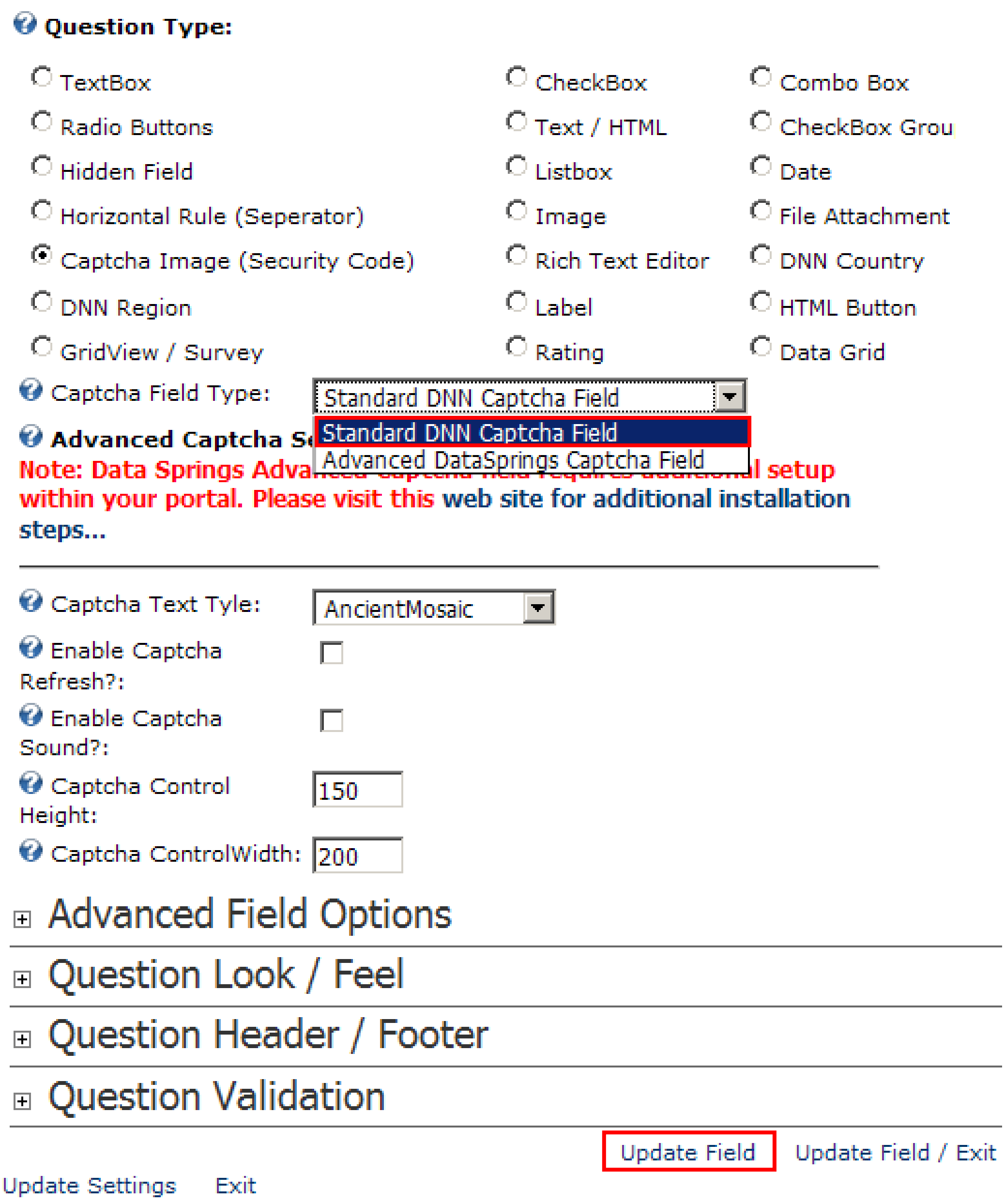
Figure 63: Creating the Captcha image

The page will be refreshed containing further options for the Captcha element i.e. the pull down menu for choosing the desired Captcha type:

- **Standard DNN® Captcha Field** – see section 7.13.1
- **Advanced DataSprings Captcha Field** – see section 7.13.2

7.13.1 Setting up a standard DNN® Captcha Field

In order to setup a standard Captcha Field, choose that option from the “Captcha Field Type” pull down menu and click on the “Update Field” link.



The screenshot below demonstrates the Captcha image element as seen by your users.

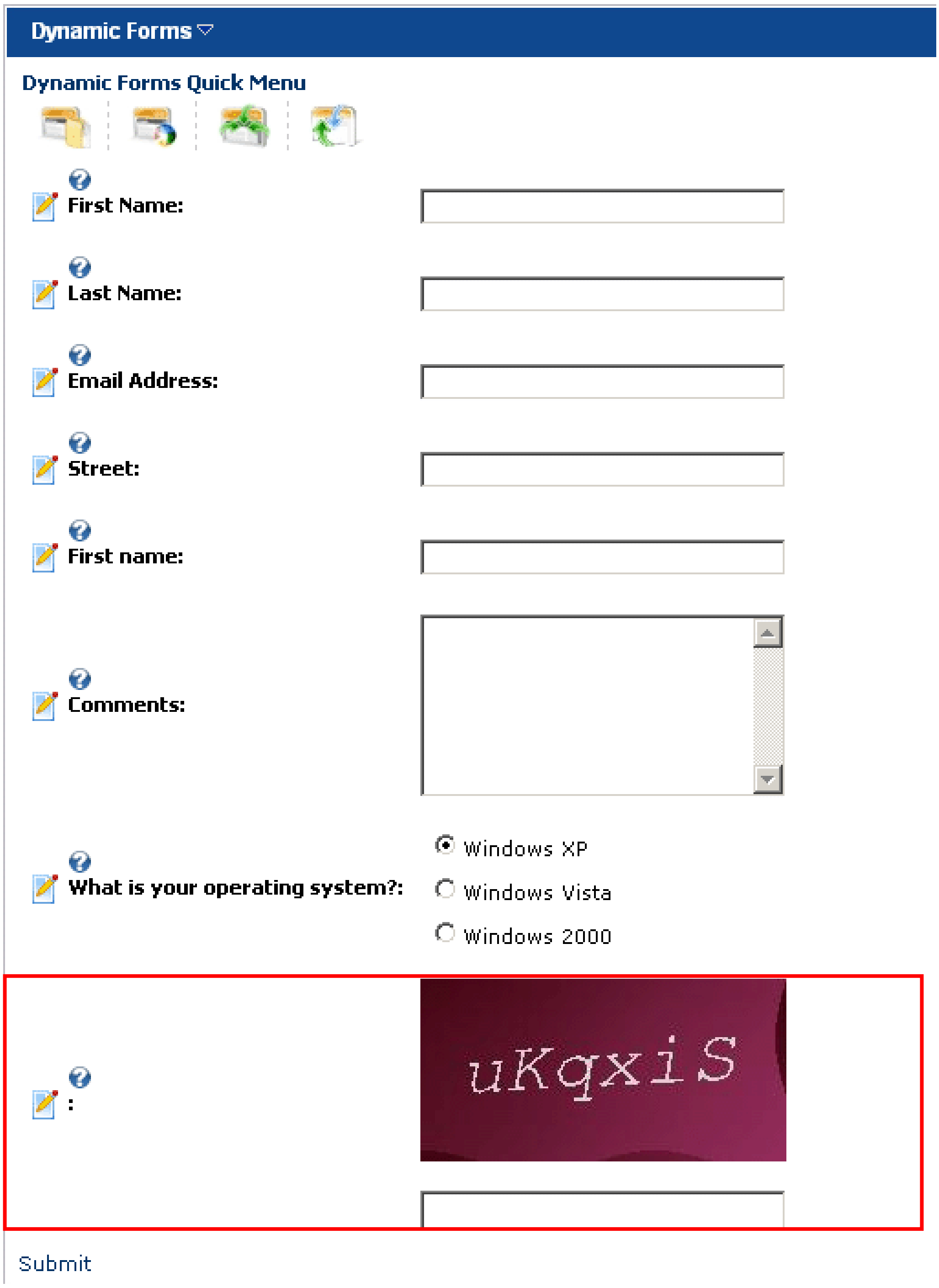


Figure 64: Example of the created Captcha image

7.13.2 Setting up an advanced Datasprings Captcha Field

In order to setup an advanced Datasprigs Captcha Field, choose that option from the “Captcha Field Type” pull down menu.

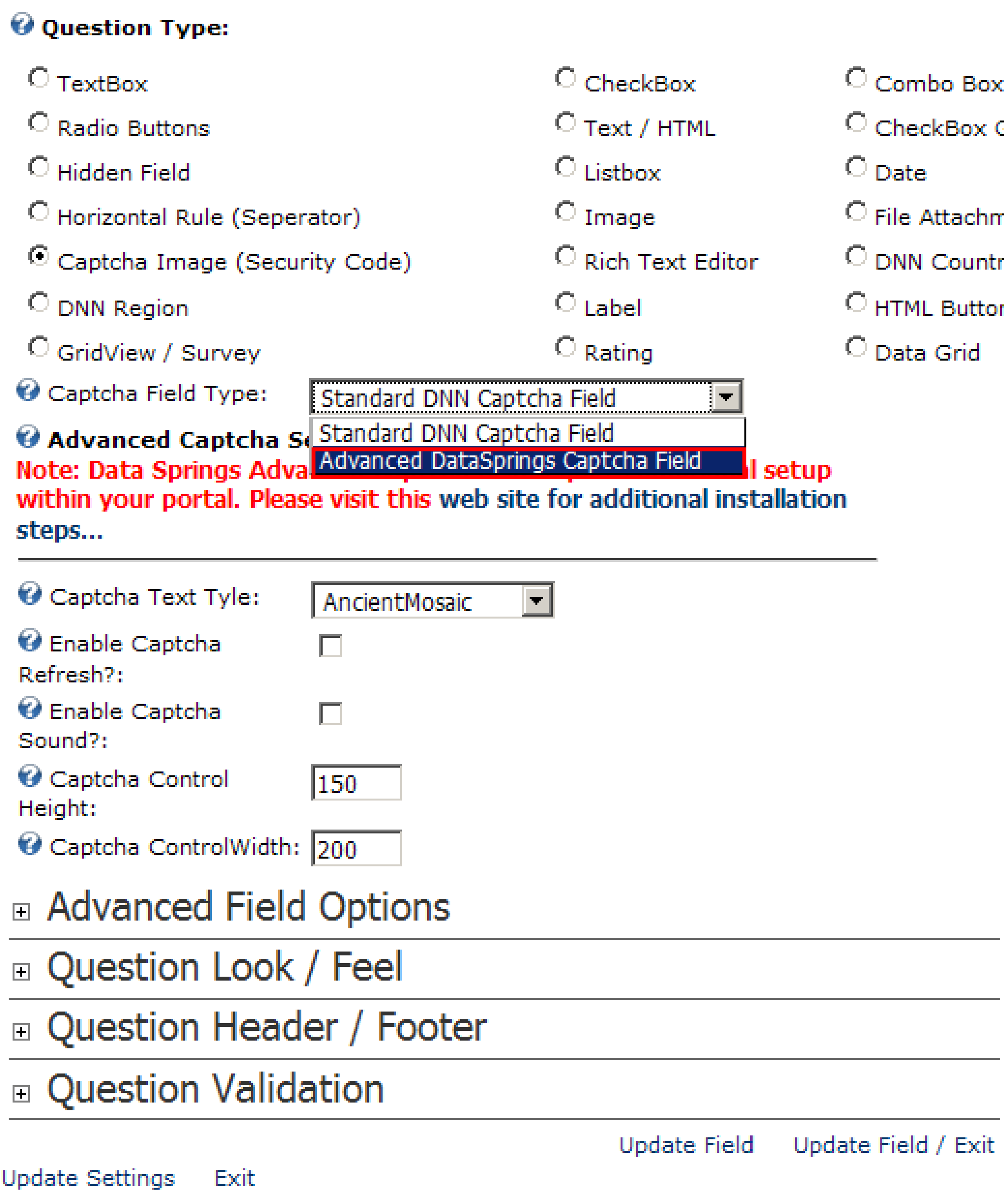


Figure 65: Setting up an advanced Datasprings Captcha Field

The following options are available:

- **Captcha Text Style** - select the desired CAPTCHA text style from this pull down menu
- **Enable Captcha Refresh** – select this option if you would like to enable the CAPTCHA refresh icon/feature within the Captcha control; this will allow the users to refresh the Captcha image i.e. change the image if they cannot make out the contents
- **Enable Captcha Sound** - select this option if you would like to enable the CAPTCHA sound icon/feature within the Captcha control which will allow your users to hear the read out of the letters and numbers within the image
- **Captcha Control Height** – use this input field to determine the height of the Captcha image in pixels
- **Captcha Control Width** – use this input field to determine the width of the Captcha image in pixels

Important note:

The Advanced Data Springs Captcha control uses a 3rd party captcha control with specific setup instructions. If you choose the Advanced Captcha field type you need to setup your DotNetNuke® Portal with these additional steps:

1. Make a backup of your web.config file.
2. Within your DotNetNuke® installation web.config file, you need to add this line to your "httpHandlers" section of the web.config. The line should be place (or can be placed) directly above the line which refers to the core DotNetNuke® Captcha field <!-- This is for CAPTCHA support -->

   <add verb="*" path="LanapCaptcha.aspx" type="Lanap.BotDetect.CaptchaHandler, Lanap.BotDetect"/>

After implementing this step your Advanced Captcha control should be configured and ready to work within Dynamic Registration or Dynamic Forms.

7.14 Creating a DNN® region

The purpose of the “DNN® Region” field is to allow you to request information about the region the user resides in. In order to start creating “DNN® Region” field, choose that option once inside the screen for creating a new dynamic field.

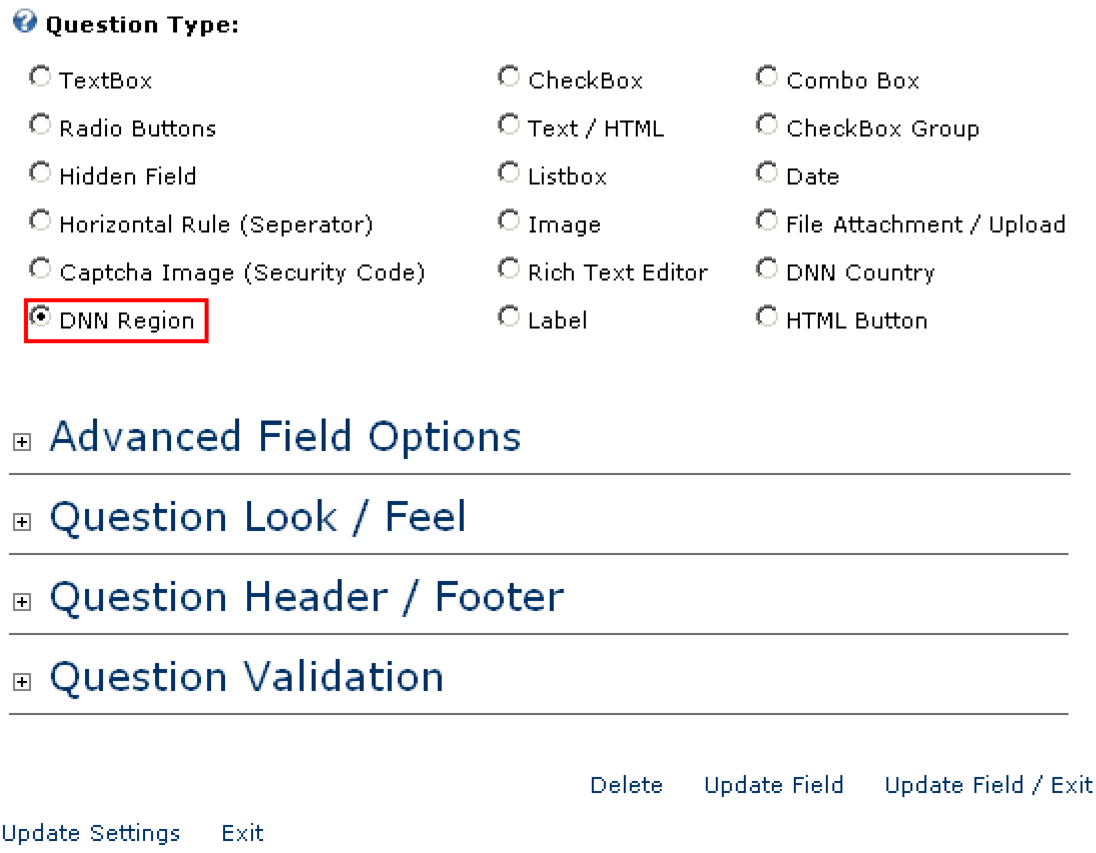


Figure 66: Creating a region

The rest of the procedure for creating the region is identical to creating the textbox.

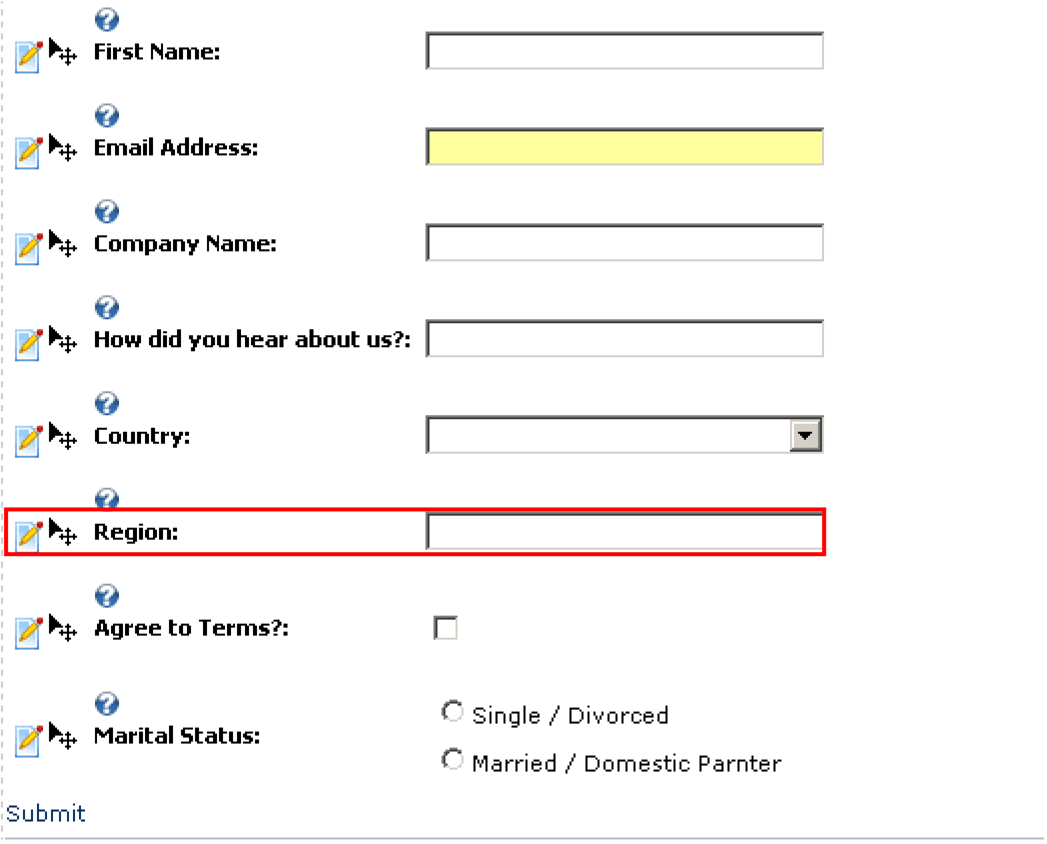


Figure 67: Example of the region element

7.15 Creating a GridView / Survey

The purpose of the “GridView/Survey” field is to allow you to create surveys. In order to start creating “GridView/Survey” field, choose that option once inside the screen for creating a new dynamic field.

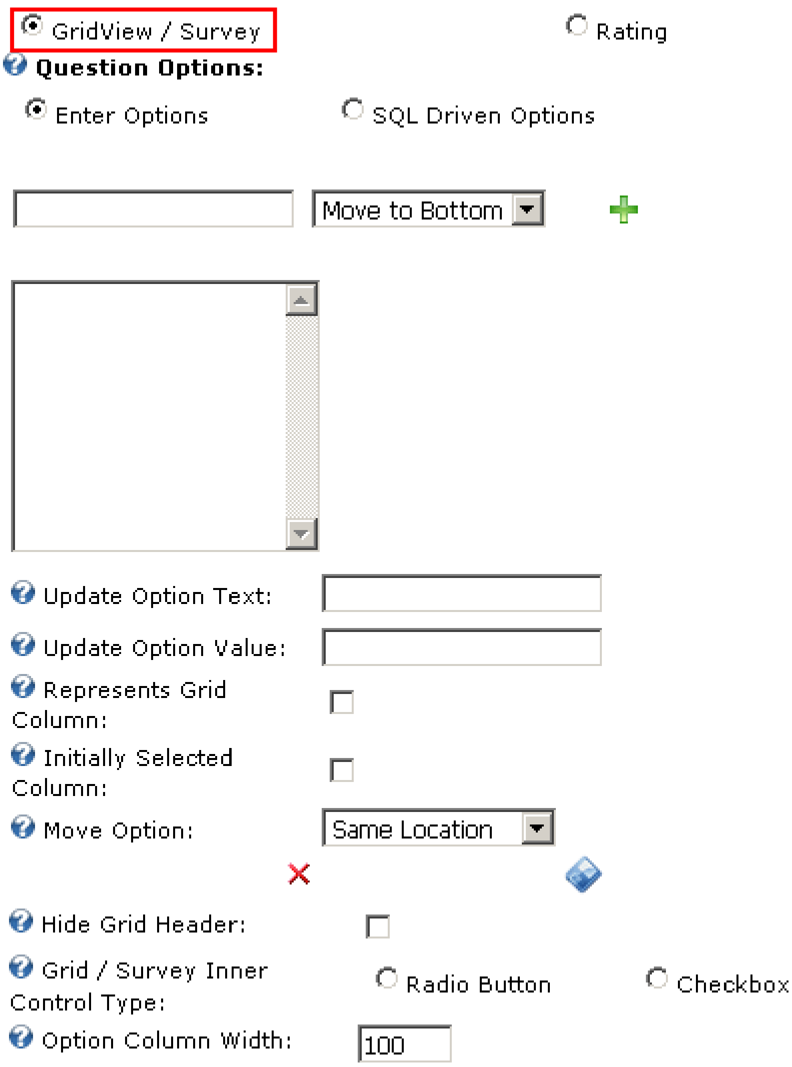
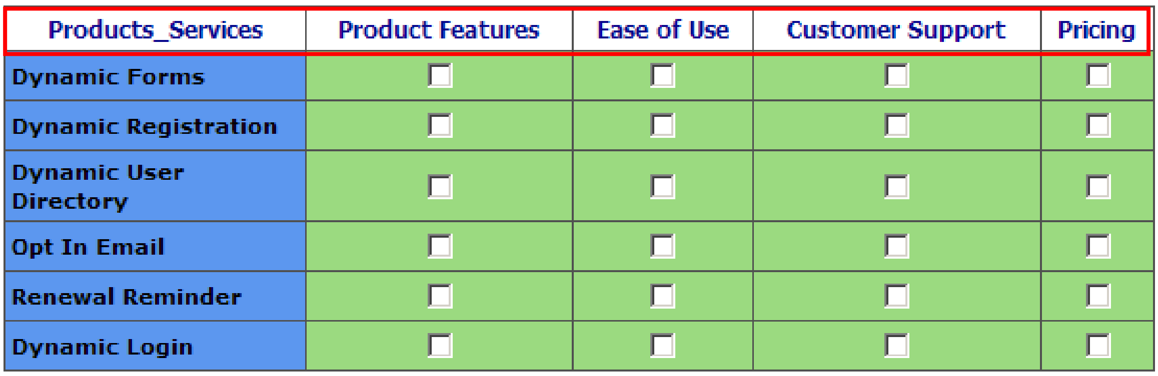


Figure 68: Creating a grid view/survey field

The following options and parameters are available:

- **Represents Grid Column** – select this option if you would like the column to be represented as a grid view initial column (see the screenshot below)



Products_Services	Product Features	Ease of Use	Customer Support	Pricing
Dynamic Forms	☐	☐	☐	☐
Dynamic Registration	☐	☐	☐	☐
Dynamic User Directory	☐	☐	☐	☐
Opt In Email	☐	☐	☐	☐
Renewal Reminder	☐	☐	☐	☐
Dynamic Login	☐	☐	☐	☐

- **Initially Selected Column** - select this option for the option you wish to be initially selected within the gridview (see screenshot below)

2a. How much do you use the modules in your collection?	Not Used	Light Use
Dynamic Forms		
Dynamic Registration		
Dynamic User Directory		
Interactive User Import		
Opt In Email		
Renewal Reminder		
News Ticker		
Flash Image Rotator		

- **Hide Grid Header** – select this option in case you would like to hide the grid view control header (see screenshot below)

Custom Module Development		
Project Implementation		
Product Training		
Site Performance Evaluation		

- **Grid/Survey Inner Control Type** - select the type of inner control that the survey / grid view will contain; this can either be a radio button for single selection implementation or a check box to allow the user to select multiple items

Custom Module Development		
Project Implementation		
Product Training		
Site Performance Evaluation		

How well are we doing?	Poor	So-So	Awesome	Blew
Customer Service				
Product Support				
Product Documentation				
Competitive Pricing				
Product Features				

Note: see section 7.18 for further information since the procedure for adding options to the survey are identical to creating combo box options.

7.16 Creating a DNN Text Suggest Field

The “DNN Text Suggest Field” allows you to create a field which will dynamically offer text suggestions to the users as they type. So while they type, they will see the suggestions allowing them to quickly make the selection and move forward.

This is done by linking a SQL query to a list of items to help the user pick an option. For example, typing "Micr" could produce suggestions such as Microbiotics, Micromanagement, Microscope, etc. This is a great alternative to offering options in a combo box.

In order to create a DNN text suggest field, choose option “DNN Text Suggest Field” inside the screen for creating a new dynamic question.

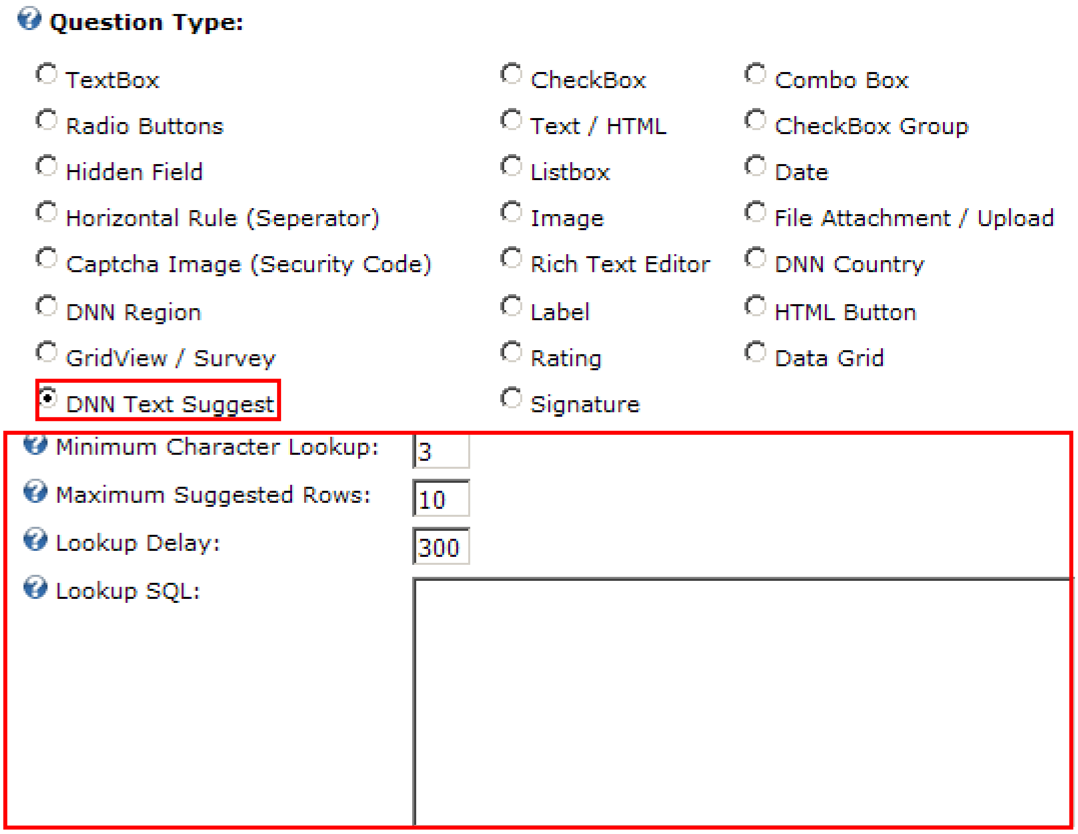


Figure 69: Creating a DNN Text Suggest Field

The following parameters are available:

- **Minimum Character Lookup** - select the minimum characters the user should type before the DNN Text Suggest starts checking the content against the lookup query
- **Maximum Suggested Rows** – specify the maximum number of rows that should be returned at a time
- **Lookup Delay** – specify the delay (in milliseconds) the user should wait while typing before the lookup occurs (**e.g.:** 300 would be for 3 seconds)
- **Lookup SQL** - enter the Lookup SQL or stored procedure SQL to return the selected list of items for the query. The query should return two columns, one called ID and another column called Name. The ID column should be unique and typically would be an integer. The Name column should return the text option.

After setting the desired parameters, click “Update Field” to save the changes and complete the procedure of creating the DNN text suggest field.

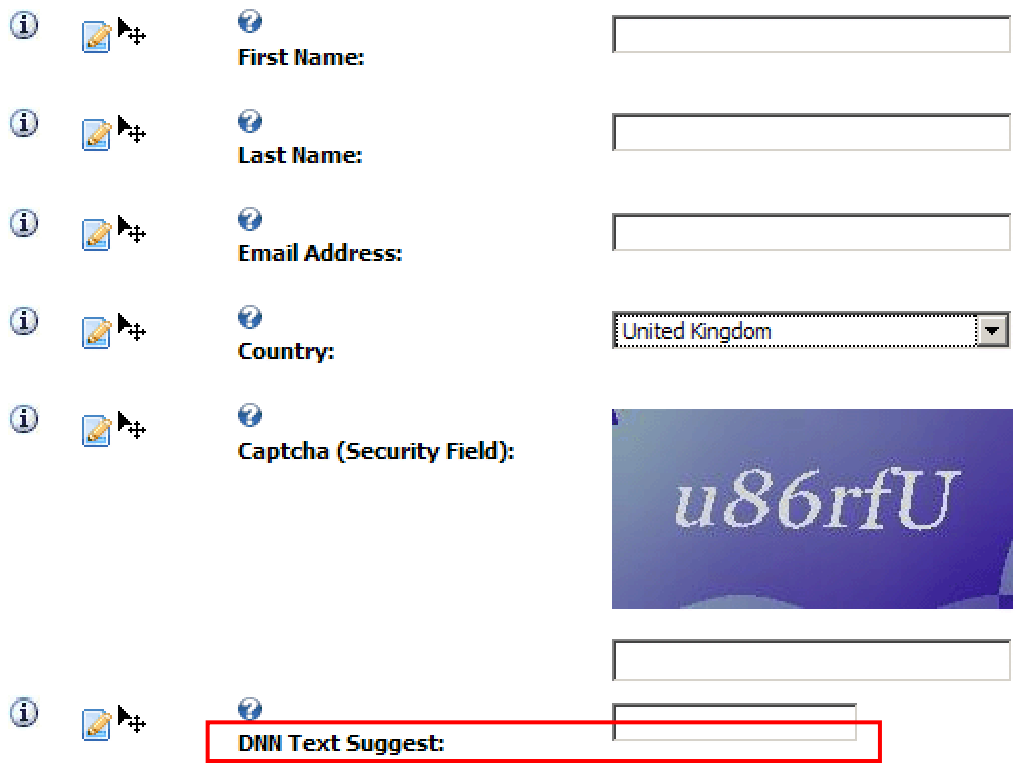


Figure 70: DNN Text Suggest Field

7.16.1.1 DNN Text Suggest Field - Example 1

This example will explain how you can use the DNN Text Suggest field to help the user by displaying all usernames from the database which begin by letters the user is typing inside the field as they type.

To create such field you will need to use the following SQL lookup:

```
select userid as ID, username as Name from users order by username
```

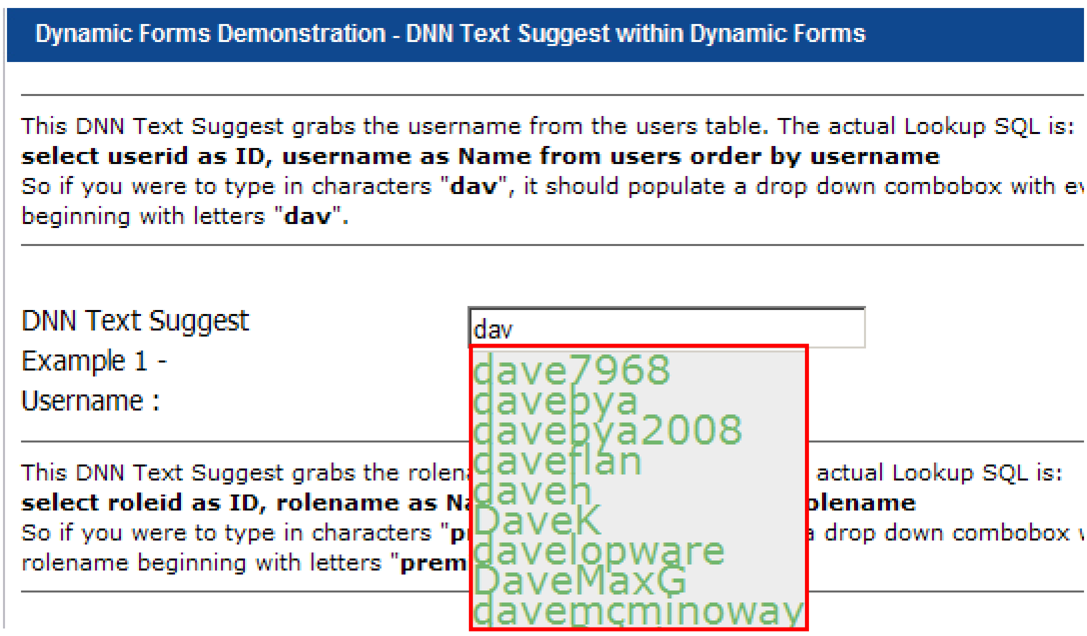


Figure 71: DNN Text Suggest Field - Example 1

As demonstrated in the screenshot, if the user types "**dav**", it should populate a drop down combobox with every username beginning with letters "**dav**".

7.16.1.2 DNN Text Suggest Field - Example 2

This example will explain how you can use the DNN Text Suggest field to help the user by displaying the user roles from database which begin by letters the user is typing inside the field as they type.

To create such field you will need to use the following SQL lookup:

```
select roleid as ID, rolename as Name from roles order by rolename
```

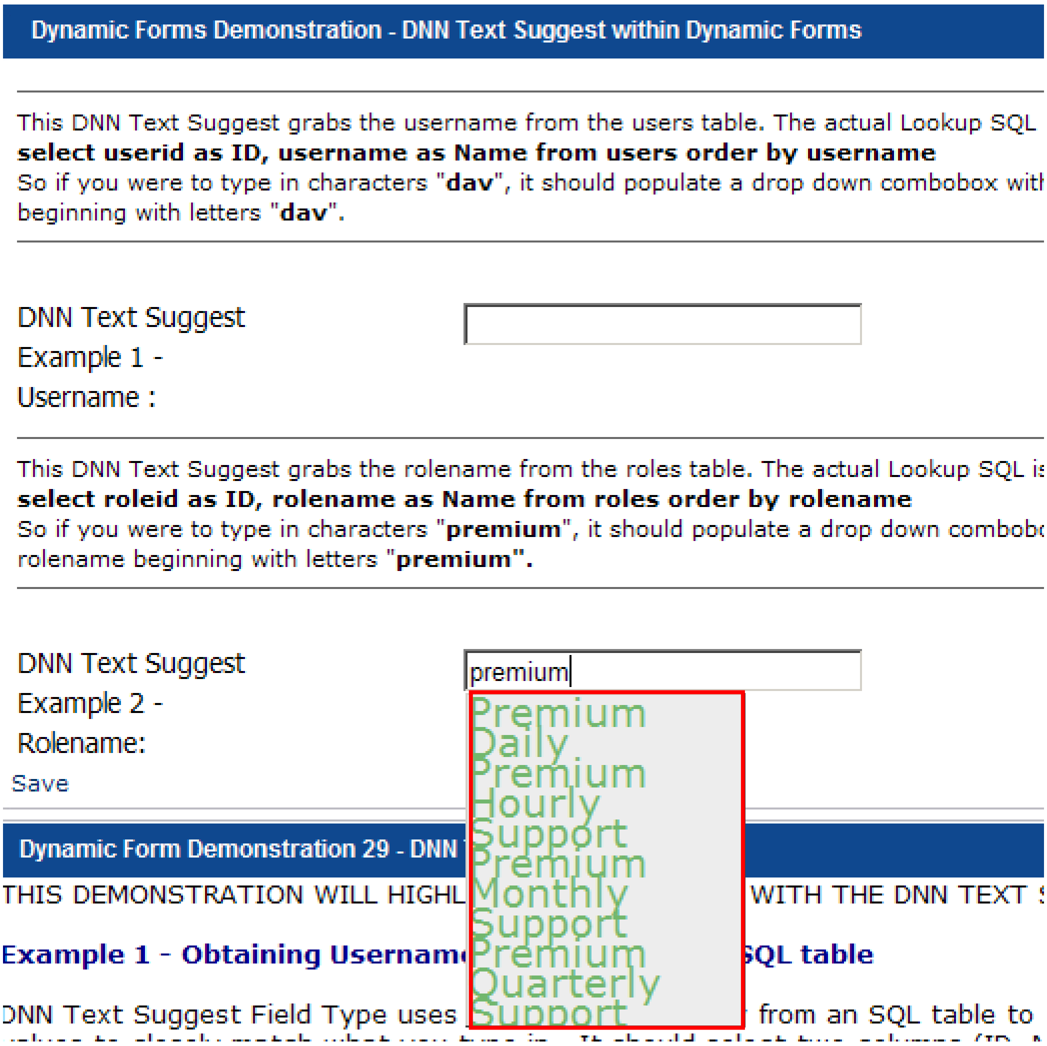


Figure 72: DNN Text Suggest Field - Example 2

As demonstrated in the screenshot, if the user types "**premium**", it should populate a drop down combobox with every role name beginning with "**premium**".

7.17 Creating a Text/HTML

The purpose of the “Text/HTML” field is to allow creating additional elements in your registration form, i.e. divide the form in sections by using various HTML elements or add additional text explaining the users what the specific part of the registration form is asking from them.

In order to start creating “Text/HTML” field, choose that option once inside the screen for creating a new dynamic field.

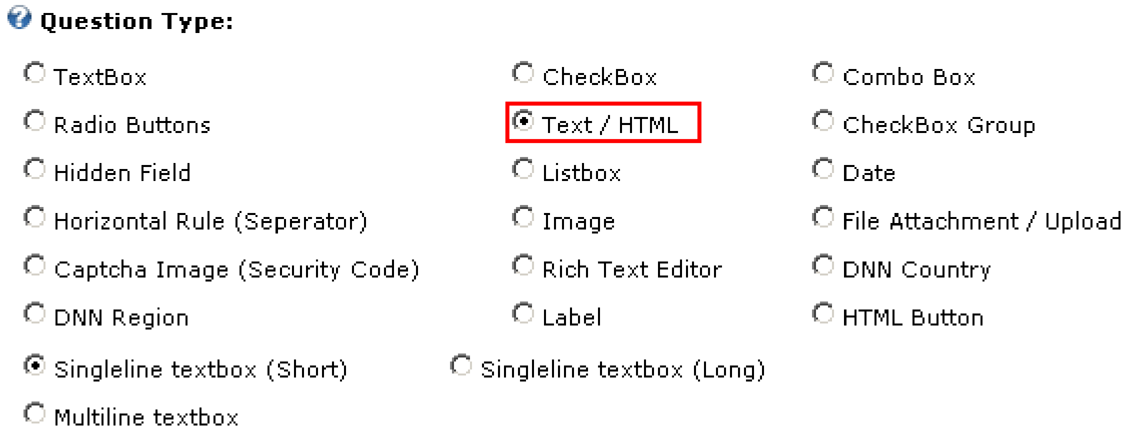


Figure 73: Creating Text/HTML element

The page will be refreshed displaying further options for this element.

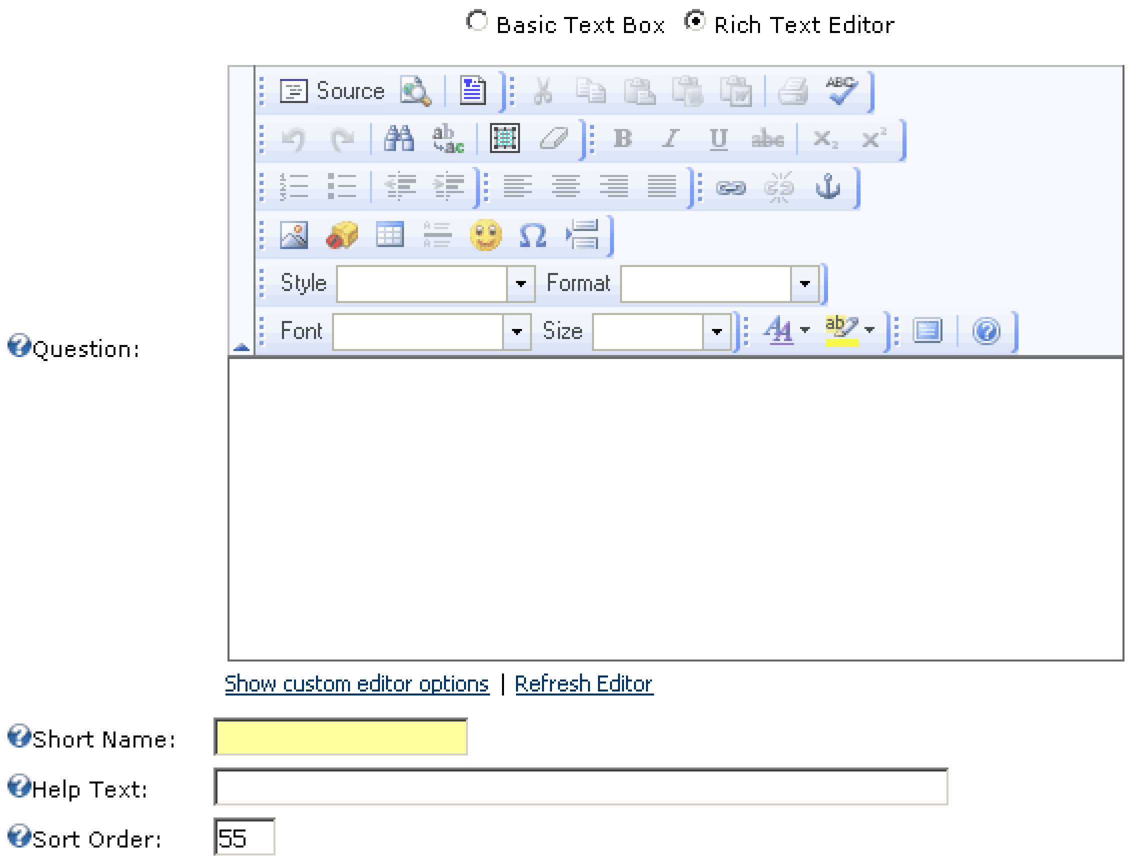


Figure 74: Creating a text/html element

The following parameters are available inside this screen:

- **Basic Text Box** – use this option if you want to define the news as text only
- **Rich Text Editor** – use this option if you want to be able to use rich formatting options (HTML, images, text formatting, etc.)
- **Paragraph** – use this pull down menu to apply the desired style (heading) to the selected text
- **Font** – pull down menu for choosing the desired font
- **Size** – pull down menu for choosing the font size
- **Color** – pull down menu for choosing the font color
- - bold
- - italic
- - underline
- - strikethrough
- - superscript
- - subscript
- - remove all formatting
- - justify left
- - justify center
- - justify right
- - justify full
- - bulleted list
- - numbered list
- - indent
- - outdent
- - cut
- - copy
- - paste
- - delete
- - undo
- - redo
- **Symbols** – pull down menu for inserting special characters
- - insert horizontal line
- - insert date
- - insert time

- - create link
- - remove link
- - insert image from gallery
- - insert table
- - preview
- - select all
- - use this option if you are pasting the text from Word so that all unnecessary Word formatting would be removed

7.18 Creating a Combo Box

In order to create a combo box element, choose option “Combo Box” inside the screen for creating a new dynamic question.

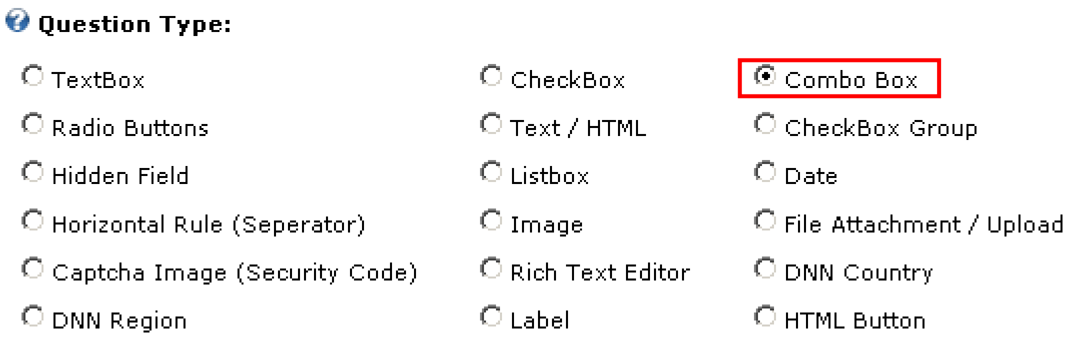


Figure 75: Creating a combo box (step 1/2)

The page will be refreshed and further options for setting up a combo box will be available.

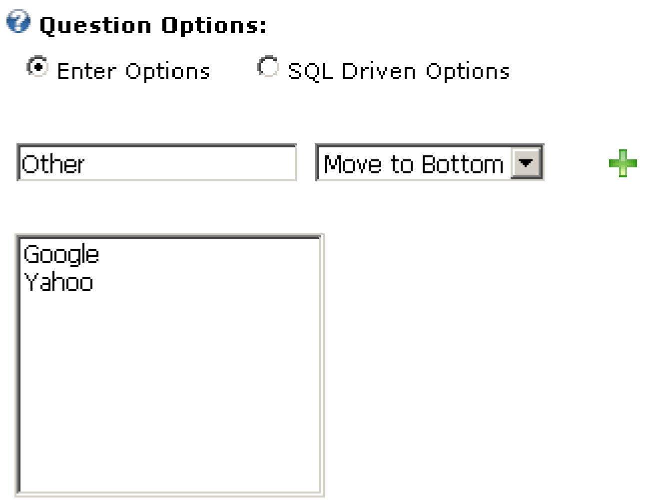


Figure 76: Creating a combo box (step 2/2)

The following parameters are available inside this screen:

- **Enter Options** – choose this option if you want to enter the options manually
- **SQL Driven Options** – option for automating process of acquiring options by extracting them from the database (see section 7.18.1)

- - button for adding the option once you've entered the option name in the input field
- - button for deleting the option; choose the desired option in the list of options and click on this button in order to delete it
- - button for updating the option
- - buttons for setting the desired sort order for the options; select the desired option and click on the up or down arrow to move the option either to the **top** or to the **bottom**
- - buttons for setting the desired sort order for the options; select the desired option and click on the up or down arrow to move the option **up or down one position**

7.18.1 Example for SQL Driven Options

Note: this option is meant for advanced users.

The "SQL Driven Options" can be used in order to simplify the process of creating options if you already have that information inside the database (**e.g.** list of states).

In order to use this functionality, choose "SQL Driven Options" and enter the query inside the text area.

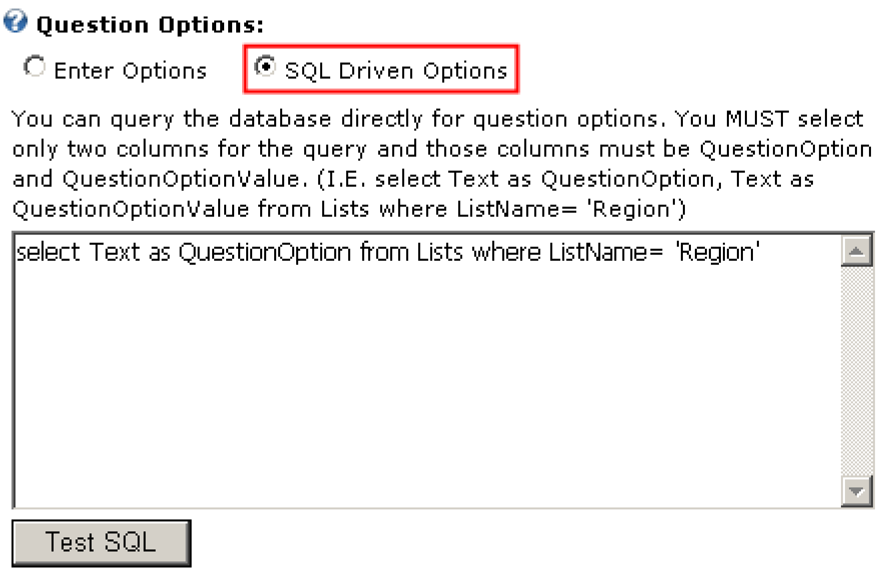


Figure 77: Using SQL Driven Options

Note: only one column should be selected for the query and that column name must be QuestionOption (i.e. select Text as QuestionOption from Lists where ListName= 'Region').

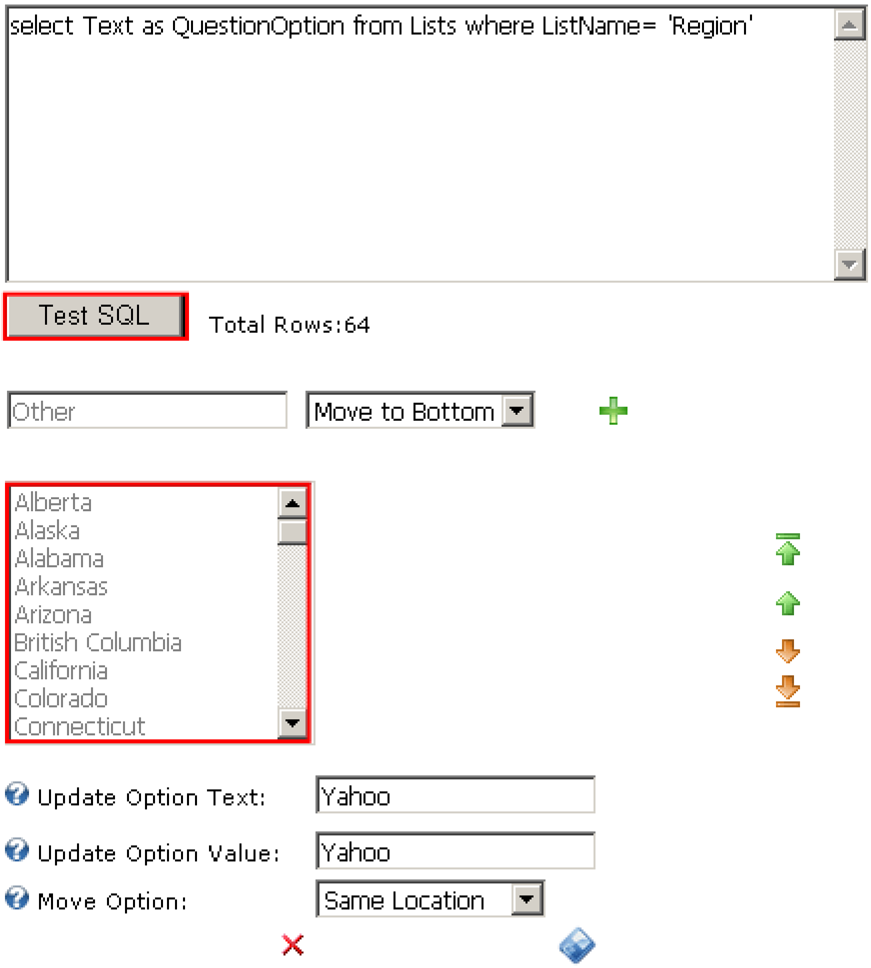


Figure 78: Using SQL Driven options

After entering the query, click on the “Test SQL” option in order to see the results of the query, i.e. whether it is obtaining desired results. The page will be refreshed and the options fill be filled in.

The last step is setting the desired default value for the pull down menu inside the part of the screen with “Advanced Field Options”.

After setting the desired parameters, click on the “Update Field” button and the procedure of creating combo box by using “SQL Driven Options” will be completed.

Note: this was one example for using “SQL Driven Options”. Advanced users can use this feature to retrieve and link fields which have options to a query instead of having to enter those values manually.

7.18.2 Example for using combo box

One example of using the combo box for registration form can be asking the user to provide information about the preferred search engine, where you can have multiple predefined answers the user will choose from (**e.g.** “Google”).

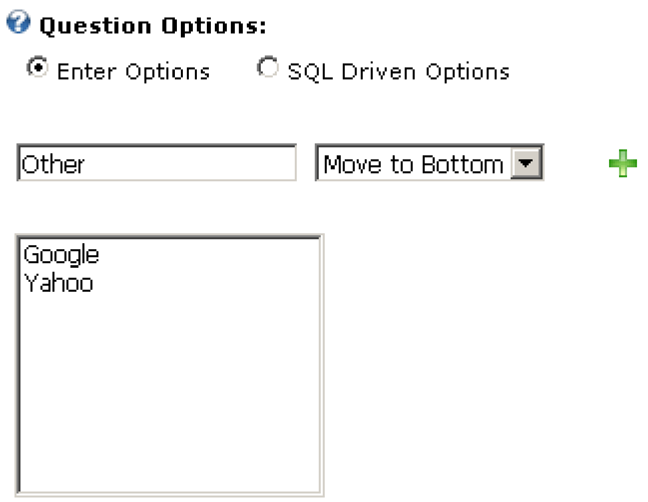


Figure 79: Example of using combo box

The parameters would be defined like this:

- **Question** – enter “What is your favorite search engine?” as a label for combo box, i.e. informing the user of the requested information
- **Question type** – choose “Combo Box”
- **Question Options** – choose “Enter Options”

After setting these parameters, enter the name of the first option inside the input field **e.g.** “Google” and click on this icon . The option will be added to the list of options. You can repeat this procedure for as many options as you like:

- Yahoo
- Google, etc

Note: use the up and down arrows to set the desired sort order and the icon to delete the option.

7.19 Creating a Checkbox

In order to start creating a checkbox element, choose option “Checkbox” inside the screen for creating a new dynamic question.

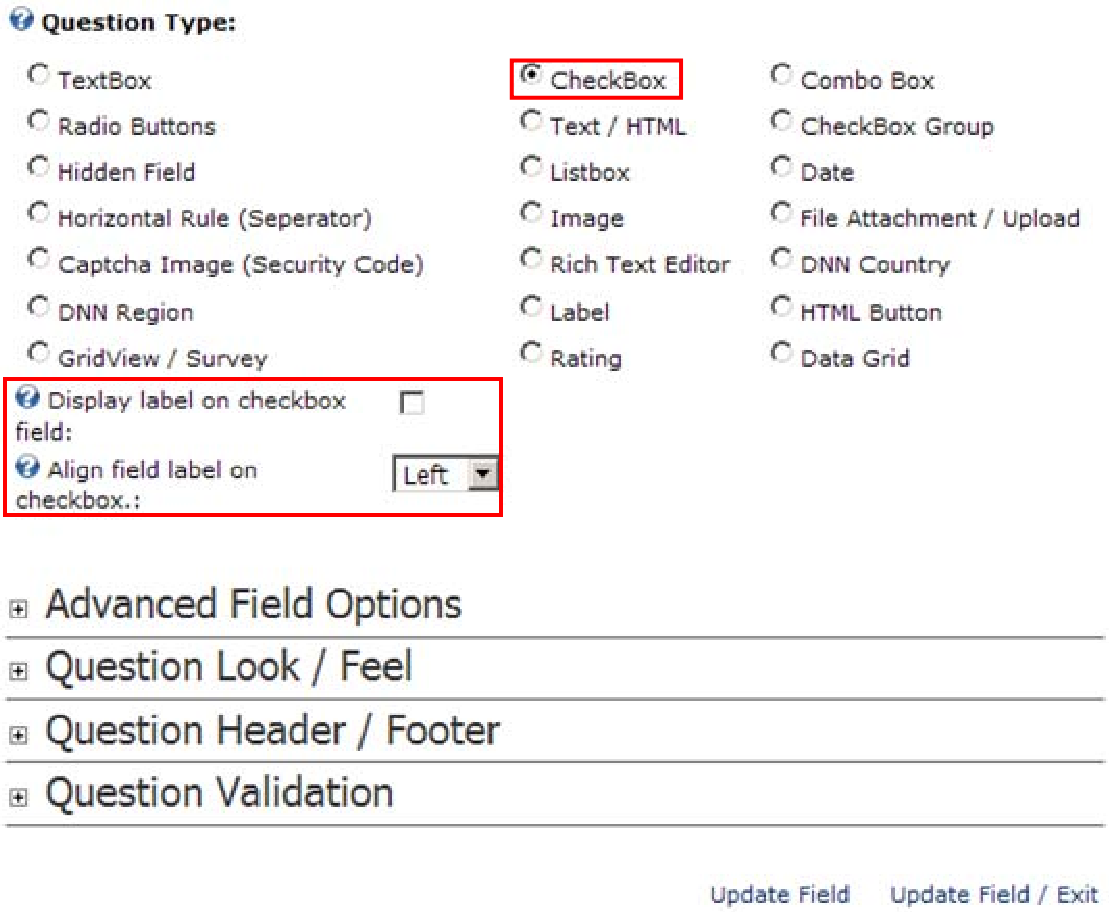


Figure 80: Creating a checkbox

After you choose the “CheckBox” option, the screen will be refreshed containing checkbox-specific options:

- **Display label on checkbox field** – select this option if you wish to display the label on the checkbox field itself (**Note:** if you do this you might want to hide the actual question label in the Question Look / Feel section).
- **Align field label on checkbox** – choose the desired alignment for the label on the checkbox field

After setting the desired parameters, click on the “Update Field” in order to complete the procedure of creating a Checkbox.

Notes:

- use the “Default Value” in order to determine if the checkbox will be selected by default or not

The following screenshot demonstrates the checkbox element as seen by the end user.

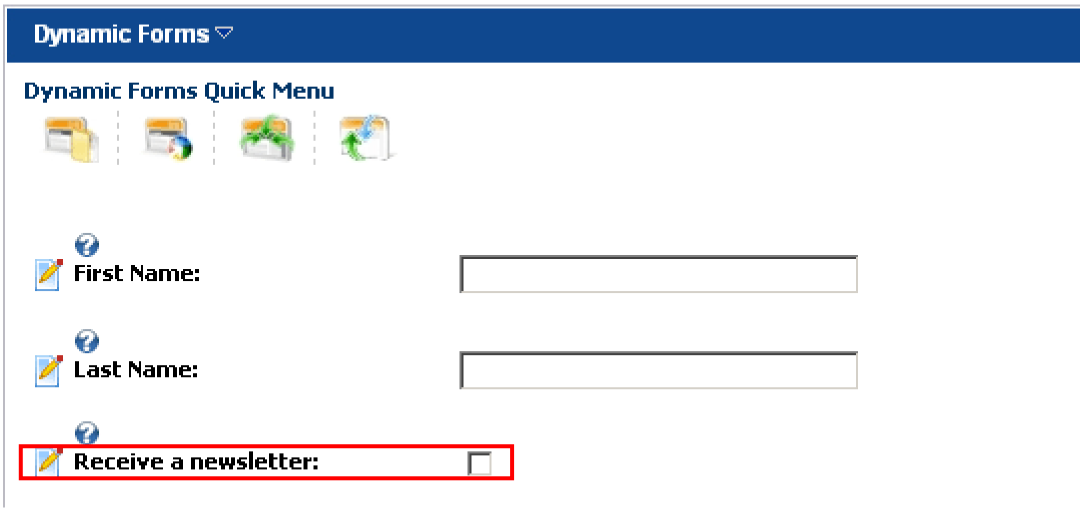


Figure 81: Example of the checkbox as seen by the end user

7.20 Creating a Checkbox Group

Checkbox group can be used when you want to acquire several answers from users by letting them select from the checkboxes. The user can select none, one or several checkboxes simultaneously.

In order to start creating a checkbox group element, choose option “Checkbox group” inside the screen for creating a new dynamic question. The page will be refreshed with further options you can set for this element.

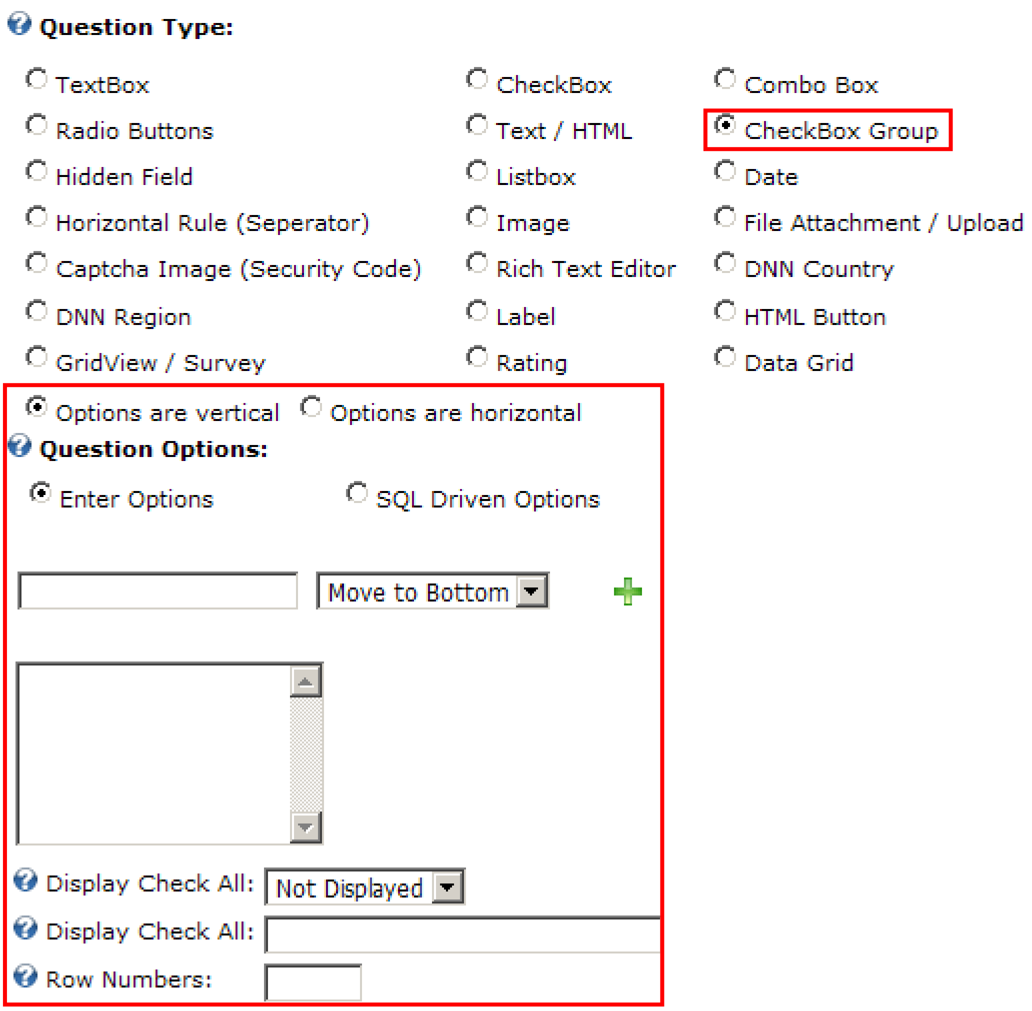


Figure 82: Creating a checkbox group

The following parameters are available inside this screen:

- **Enter Options** – choose this option if you want to enter the options manually
- **SQL Driven Options** – use this option in order to automatically insert options from the database, if you already have them defined so that you wouldn't have to do it manually (see section 7.18.1)
- - button for adding the option once you've entered the option name in the input field
- - button for deleting the option; choose the desired option in the list of options and click on this button in order to delete it
- - buttons for setting the desired sort order for the options; select the desired option and click on the up or down arrow to move the option either to the **top** or to the **bottom**
- - buttons for setting the desired sort order for the options; select the desired option and click on the up or down arrow to move the option **up or down one position**
- **Display Check All** – use this pull down menu to decide if you wish to display the "check all" and if so the location of the feature i.e. above or below the check box list
- **Display Check All** – enter the desired text for the "Check all" feature. i.e. "Select All"
    - **Note:** If no text is specified then the module will render a localized variable for this setting
- **Row Numbers** – Row numbers is a feature used to select the number of items to display per row. The row numbers property works directly with the horizontal/vertical alignment property. For example, although the alignment might be setup for horizontal do you want it to display 2 across before starting a new row? If so select 2 for the row numbers property.

7.20.1 Example for using checkbox group

One example of using the checkbox group for registration form can be asking the user to provide information about your products, where you can have multiple predefined answers as checkboxes (**e.g.** “Which products do you like?”, and then you can list of all your products as checkboxes/answers).

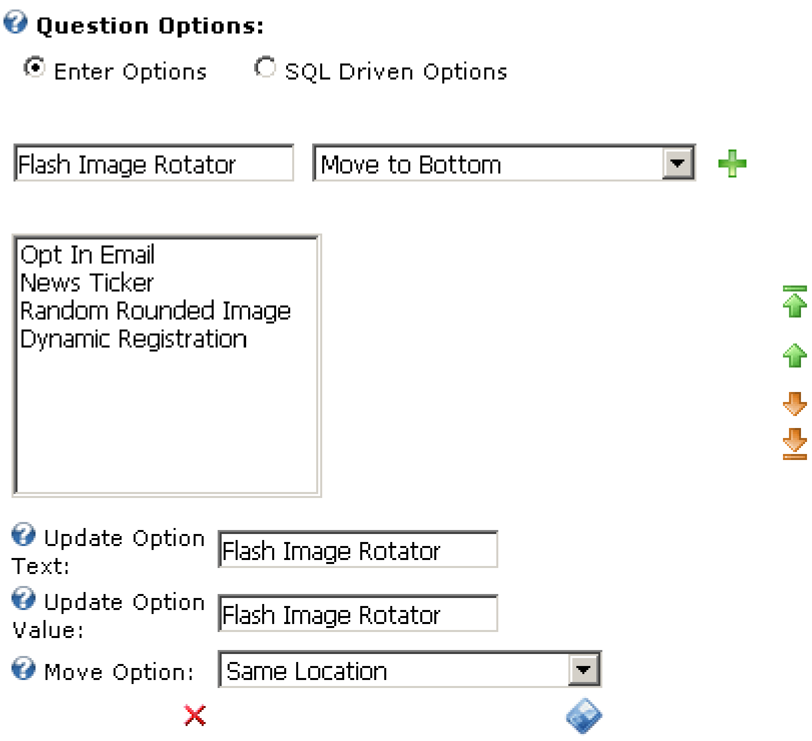


Figure 83: Example of using combo box

The parameters would be defined like this:

- **Question** – enter “Which products do you like?” as a label for checkbox group, i.e. informing the user of the requested information
- **Question type** – choose “Checkbox group”
- **Question Options** – choose “Enter Options”

After setting these parameters, enter the name of the first option inside the input field **e.g.** “Image Flash Rotator” and click on this icon . The option will be added to the list of options. You can repeat this procedure for as many options as you like:

- News Ticker
- Quick Poll, etc

Note:

- use the up and down arrows to set the desired sort order and the icon to delete the option.
- after setting the desired options for the checkbox group, you can click on the “+” symbol next to the “Advanced Field Options” in order to set the default value i.e. checkbox that will be selected by default in your registration form; you can select more than one checkbox to be checked by default

7.21 Creating a Listbox

In order to start creating a listbox element, choose option “Listbox” inside the screen for creating a new dynamic question.

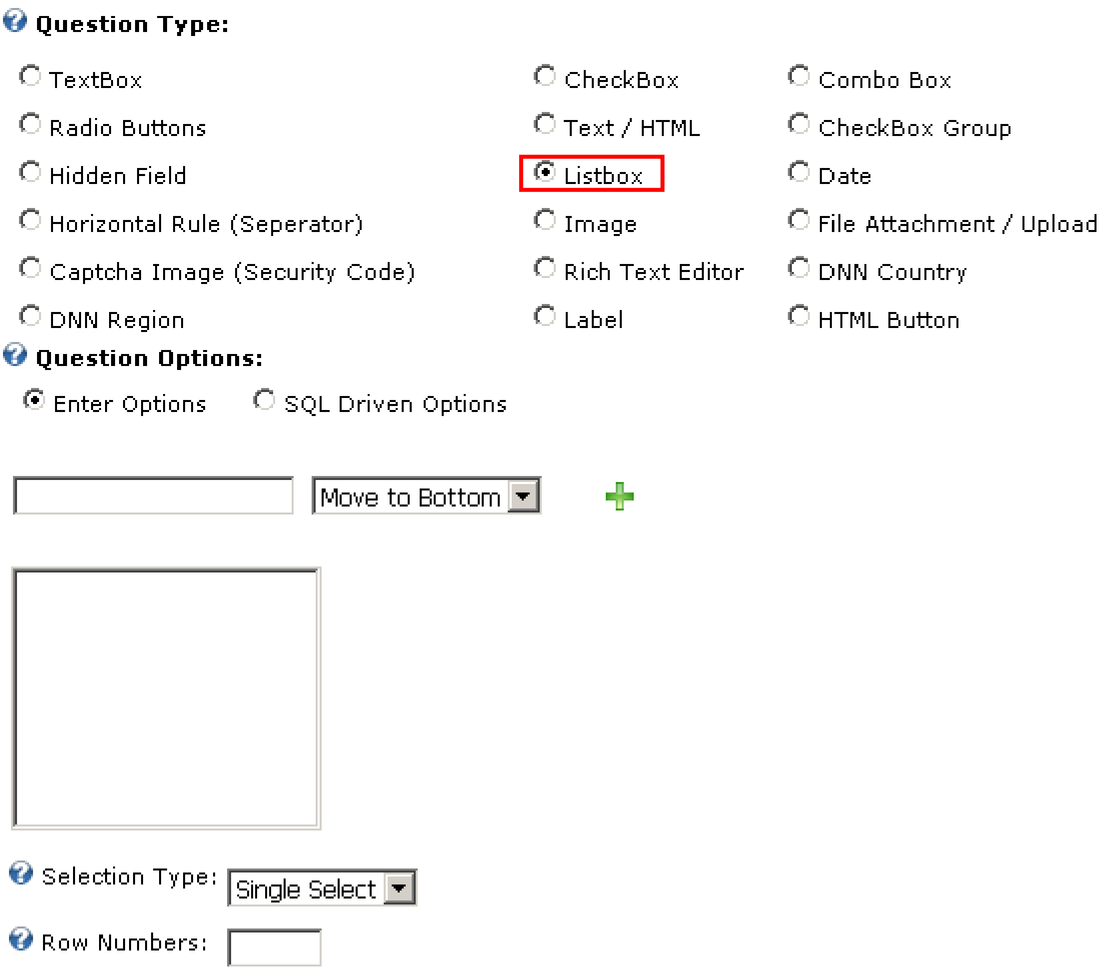


Figure 84: Creating a listbox

After choosing the “listbox” option, click on the “Update Field” link and choose the newly created element from the pull down again.

The rest of the procedure for creating listbox options is identical to creating radio buttons (see section [7.11](#)). The screenshot below illustrates a created listbox as seen in the front end.

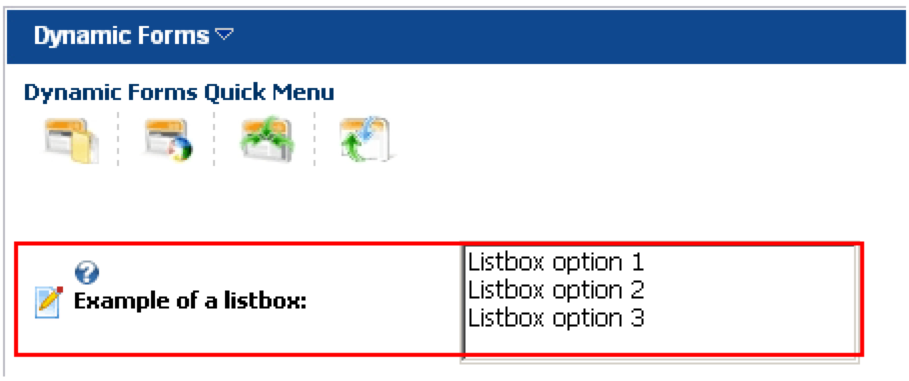


Figure 85: Example of the created listbox

7.22 Creating an Image Element

The purpose of the “Image” element is to allow users to upload images. In order to start creating an image element, choose option “Image” inside the screen for creating a new dynamic question. The screen will be refreshed containing further image specific parameters.

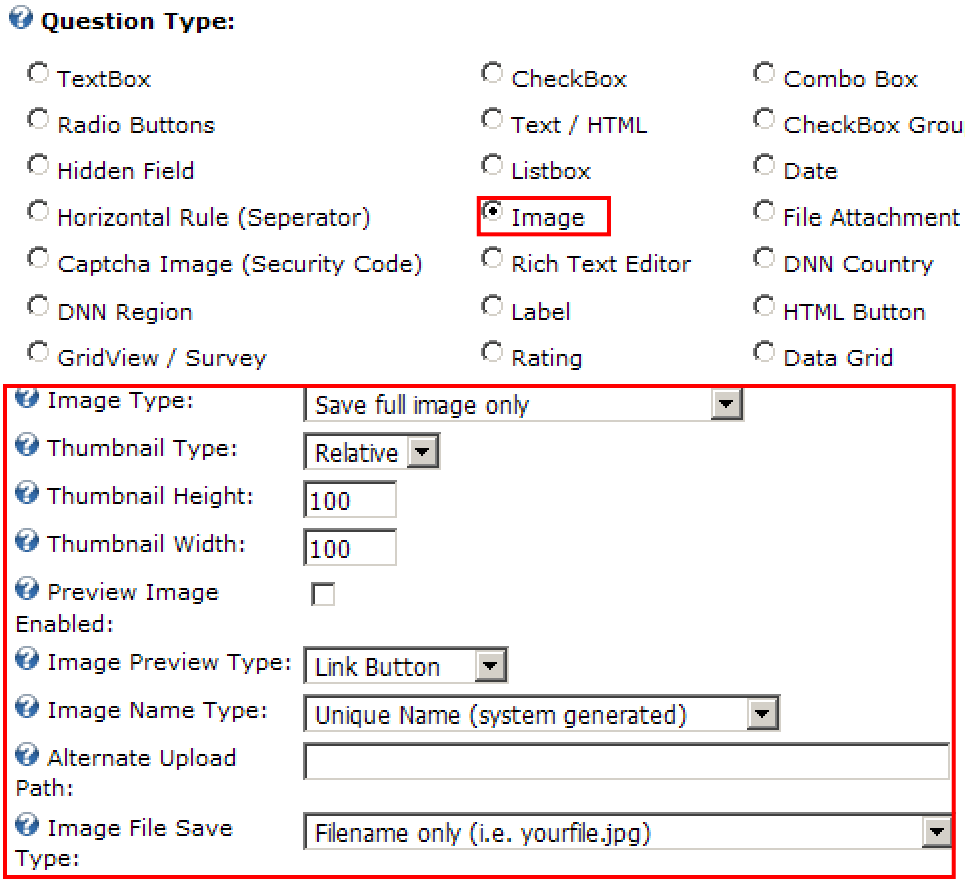


Figure 86: Creating an image element

The following parameters are available inside this screen:

- **Image Type** – choose the desired image type from this pull down menu
  - **Save Full Image Only** – select this option if you want to display the full image uploaded by the user
  - **Save Thumbnail Only** – select this option if you want to display a thumbnail image of the image uploaded by the user
  - **Save Thumbnail and Full Image Link** – select this option if you want to display a thumbnail of the image uploaded by the user with a link for opening a full image
- **Thumbnail Type**
  - **Relative** - if you choose to have it created based on relative parameters, the thumbnail will be created based on width and height of the image the user is uploading
  - **Fixed** - if you choose to have it created as fixed, the thumbnail image will always be generated based on the thumbnail width and height
- **Thumbnail Height** – set the desired height for the thumbnail image (in pixels, **e.g.** 100)
- **Thumbnail Width** – set the desired width for the thumbnail image (in pixels, **e.g.** 100)
- **Preview Image Enabled** – select this checkbox if you want to allow your users to see the preview of the image

- **Image Preview Type** – use this pull down menu to specify the method for enabling the image preview; you can select either to have a link button appear, an image, or automatically as soon as the user selects the file
  - o **Image Name Type** – use this pull down menu to specify the method for naming the images; this can either be a unique name, a friendly name which is the name of the file with a date and time stamp, or the exact filename of the file
- **Alternate Upload Path** – Please specify an alternate portal upload directory for this image. By default all files will be uploaded under a directory called DynamicForms_Images within your portal root directory (usually /portals/0/DynamicForms_Images/).
- **Image File Save Type** – use this pull down menu to specify the method for naming the file i.e. saving it in the database, this can either be just the name of the file or it can be the full HTTP path to the file.

After setting the desired parameters, click on the "Update Field" link in order to save the changes. The screenshot below demonstrates the image element as seen by the end users.

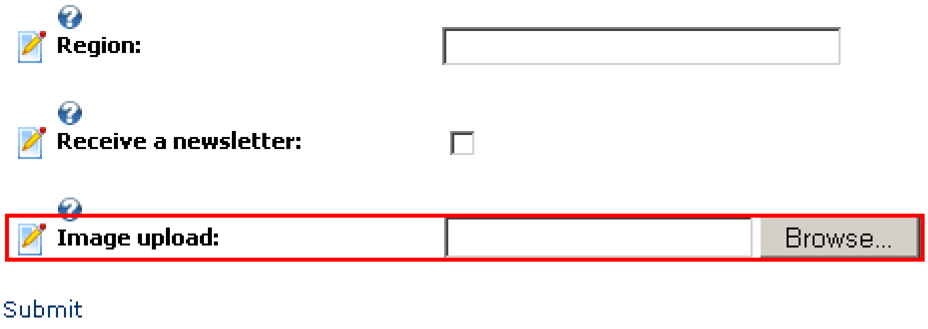


Figure 87: Image element as seen by the end users

7.23 Creating a Rich Text Editor

The purpose of the "Rich Text Editor" option is to allow your users to use the text editor in the front end. In order to start creating the rich text editor element, choose option "Rich Text Editor" inside the screen for creating a new dynamic question.

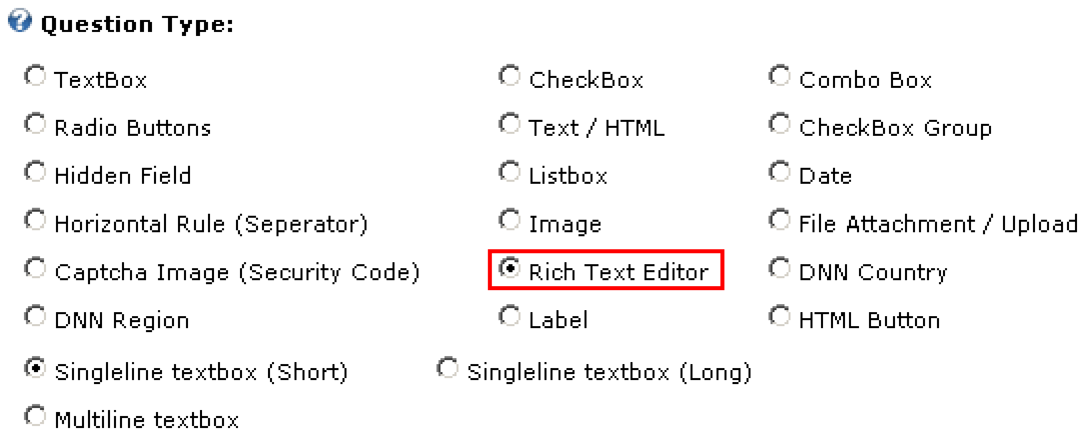


Figure 88: Adding the rich text editor element

After selecting the "Rich Text Editor" radio button, click on the "Update Field" link in order to save the changes and complete the procedure of adding the "Rich Text Editor". The screenshot below demonstrates the added rich text editor as seen by the end users.

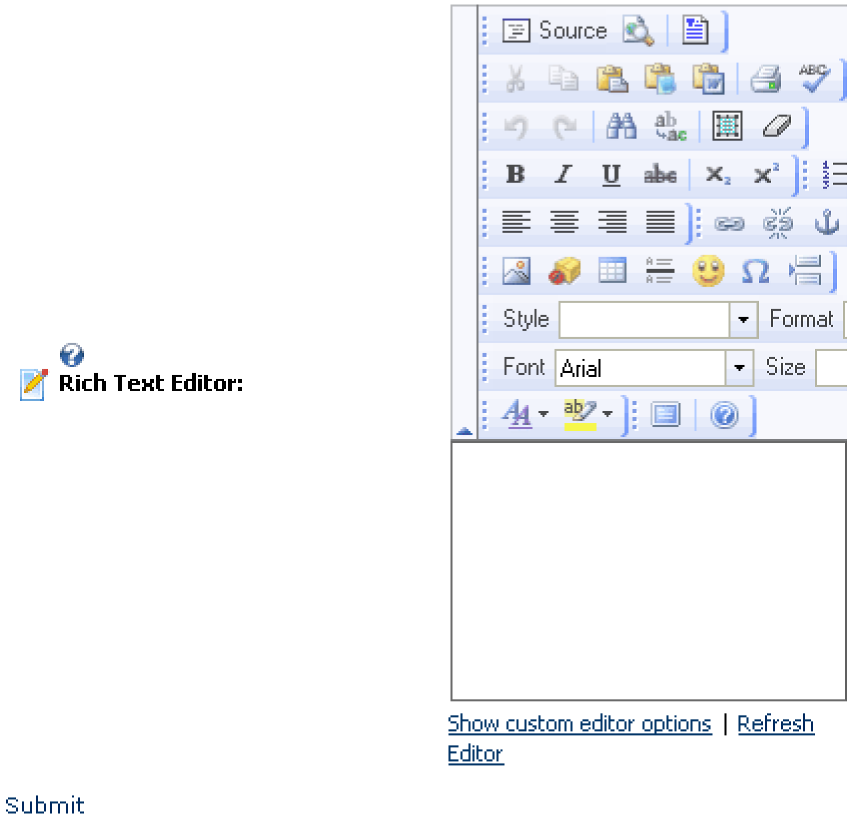


Figure 89: The "Rich Text Editor" as seen in the front end

7.24 Creating a Label

The purpose of this field is to allow you to create an independent label within the form. In order to create a label field, choose that option from the “Question type” part of the screen while creating a new field.

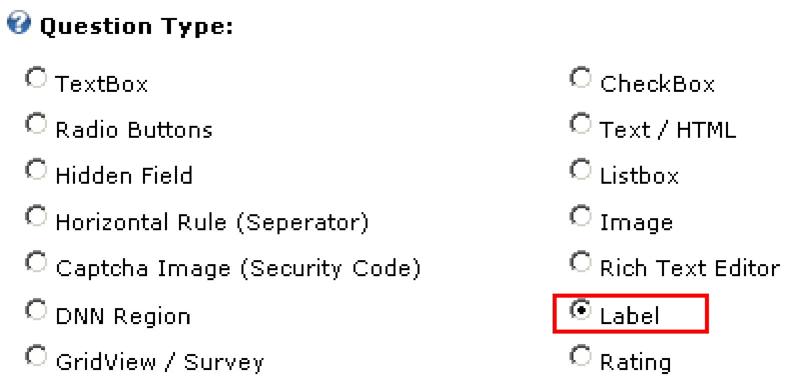


Figure 90: Creating a label

After selecting the “Label” option, click the “Update Field” link and the label will be added. The screenshot below demonstrates a label field.

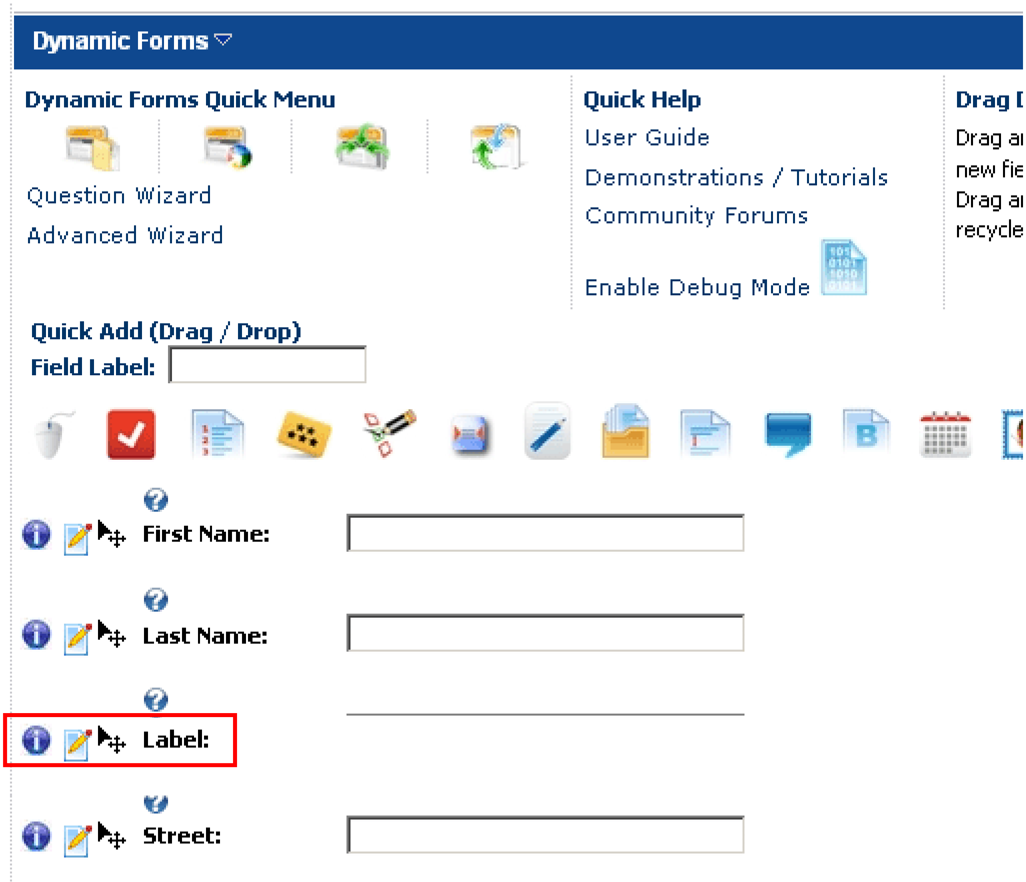


Figure 91: Example of the label field

7.25 Creating a rating

The purpose of this field is to allow you to create a rating field which will allow your visitors to give a rating on the desired subject by choosing the desired amount of stars. In order to create a label field, choose that option from the “Question type” part of the screen while creating a new field.

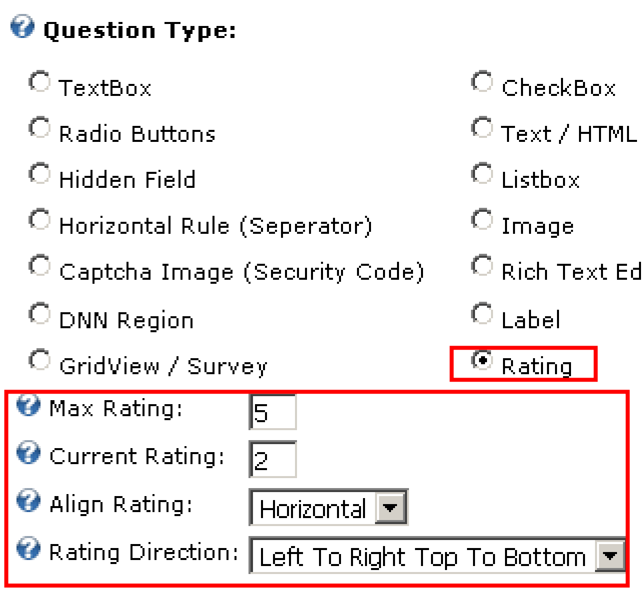


Figure 92: Creating a rating

The following options and parameters are available:

- **Max Rating -** select the maximum rating that the module should allow for the field
- **Current Rating** - select the rating that the module should initially display for the field
- **Align rating** – choose the desired alignment for the rating; this setting can either be horizontal or vertical
- **Rating Direction** - select the desired direction for the rating control; this setting can either be left to right top to bottom, or right to left bottom to top

The screenshot below demonstrates a rating field.

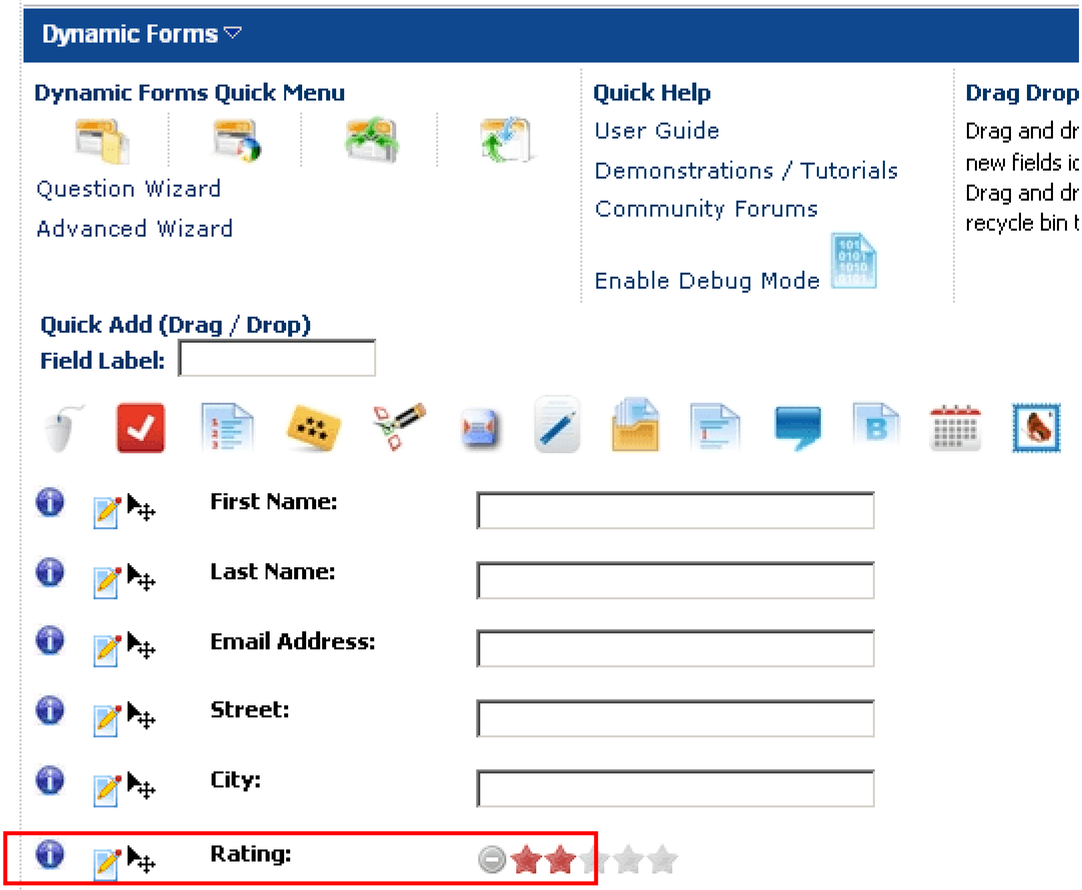


Figure 93: Example of the rating field

7.26 Creating a Date

The “Date” field is used for acquiring a date from the user, either by asking them to select one from the pull down menus or by entering it manually (**note**: depending on the selected type of date – further discussed below). In order to start creating a date, choose option “Date” while creating a new dynamic field.

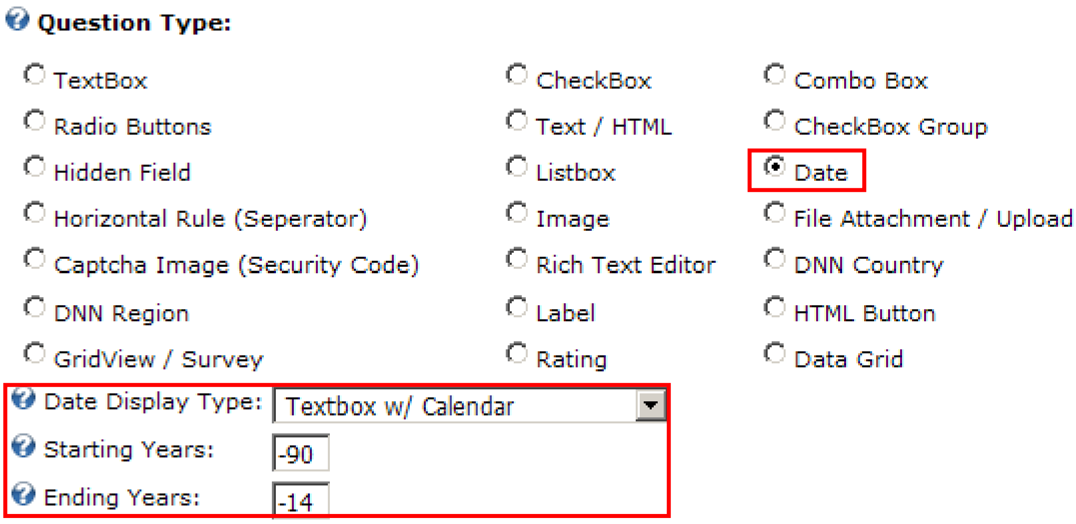


Figure 94: Creating a "Date" dynamic field (step 2/2)

The page will be refreshed with further options for selecting the date type (format). The following options are available:

- **Date Display Type** – the following options are available inside the “Date Display Type” pull down menu:
  - **Textbox w/Calendar** – see section 7.26.1
  - **Month and Day** – see section 7.26.2
  - **Month and Year** – see section 7.26.3
  - **Month, Day, Year (Textbox)** – see section 7.26.4
- **Starting Years** - please specify the number of years to go back, for example a -90 would start the years for the date field to start 90 years ago
  - **Note** :the starting years property is only valid for date field types that use a years dropdownlist
- **Ending Years** - please specify the number of years from now to add to the years dropdownlist, for example a -5 would start the years for the date field to stop 5 years ago.
  - **Note**: the starting years proprety is only valid for date field types that use a years dropdownlist

After setting the desired parameters, click the “Update field” link in order to complete the procedure of creating a “Date” dynamic field.

7.26.1 Textbox with Calendar

The following screenshots demonstrates the “Textbox and Calendar” date type as seen by the end users. The user will be able to either enter the date manually or choose one from the calendar by clicking on the “Select Date” link.

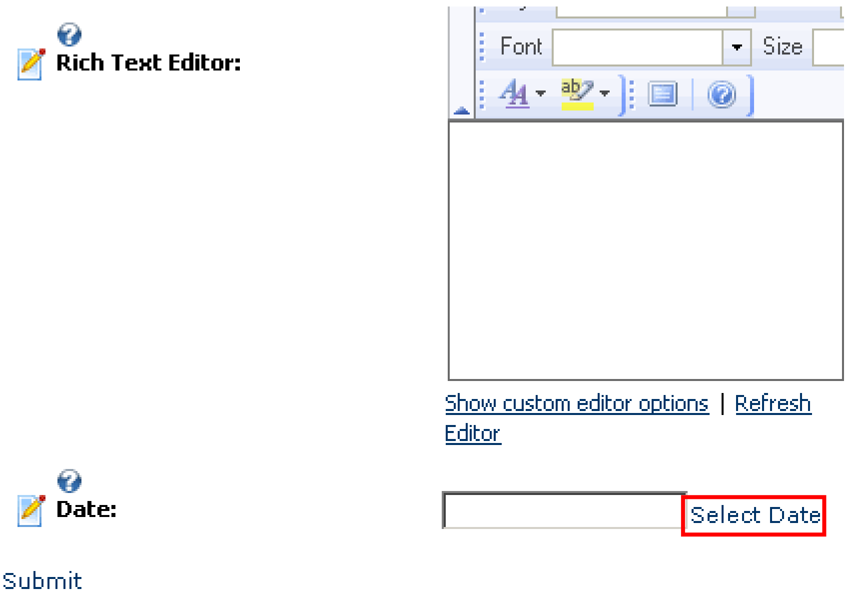


Figure 95: Textbox with Calendar

Figure 96: Choosing the date from the calendar

7.26.2 Month and Day

The following screenshots demonstrates the “Month and Day” date type as seen by the end users. The user will be able to enter the date by using the month and day pull down menus.

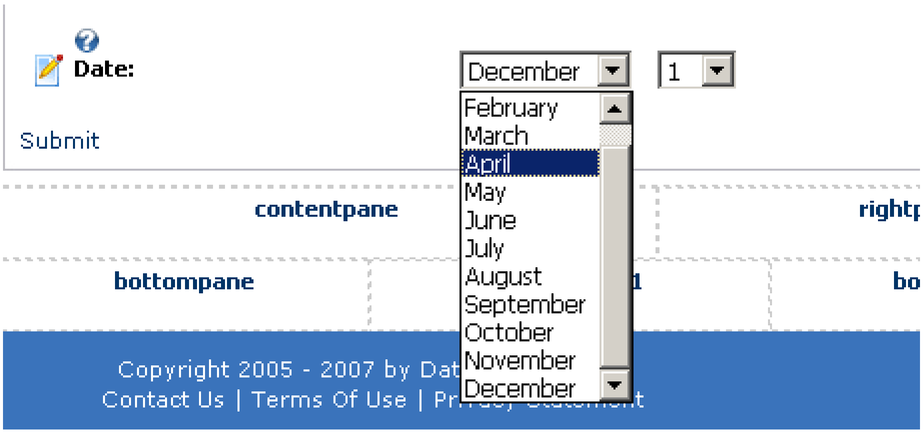


Figure 97: “Month and Day” example

7.26.3 Month and Year

The following screenshots demonstrates the “Month and Year” date type as seen by the end users. The user will be able to enter the date by using the month and year pull down menus.

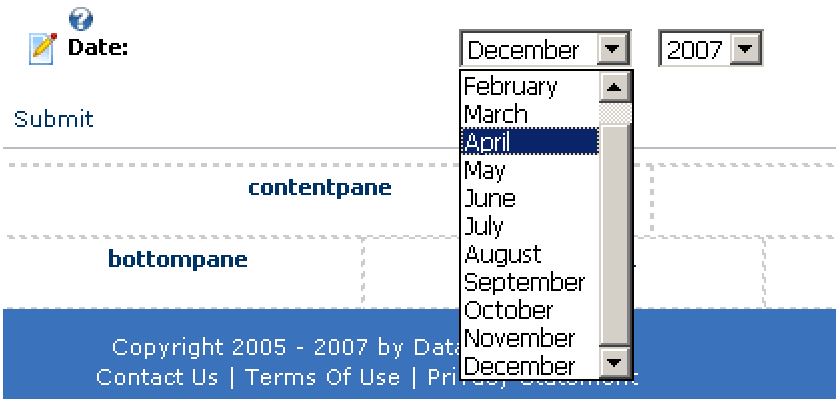


Figure 98: “Month and Year” example

7.26.4 Month, day, year textbox

The following screenshots demonstrates the “Month, day, year” date type as seen by the end users. With this date type the user enters the date manually.

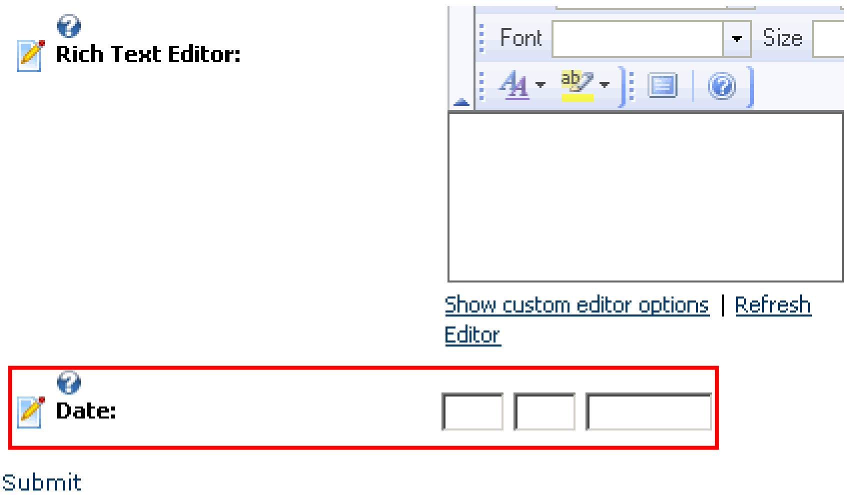


Figure 99: “Month, day, year textbox” example

7.27 Creating a File Attachment/Upload

The purpose of the “File Attachment/Upload” element is to allow your users to upload files via form. In order to start creating a file attachment element, choose option “File Attachment/Upload” inside the screen for creating a new dynamic question.

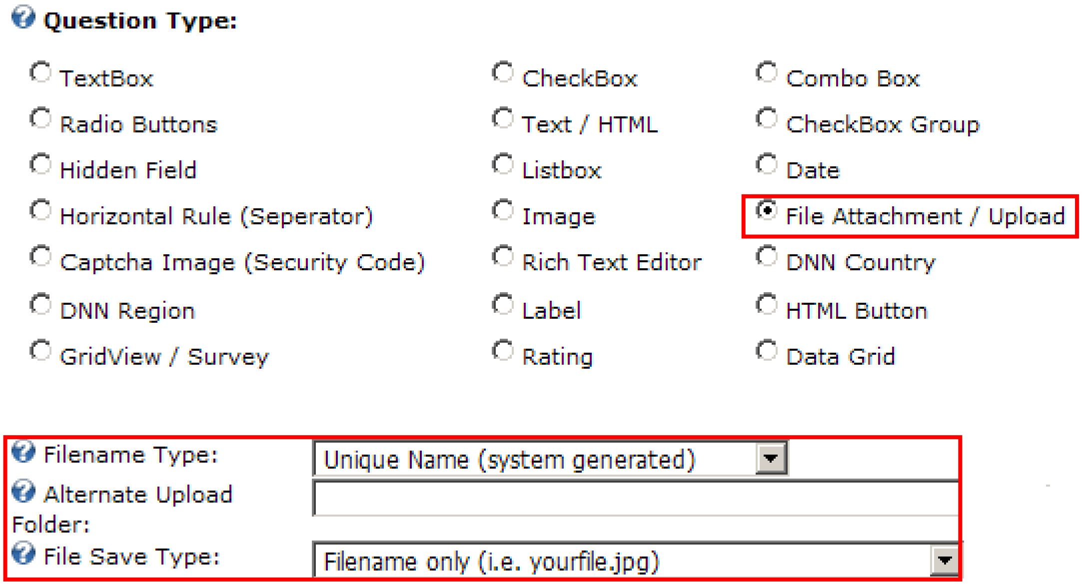


Figure 100: Creating a “File Attachment/Upload” element

The following options are available:

- **Filename Type** – use this pull down menu to specify the method for storing the name of the file, this can either be a unique name, a friendly name which is the name of the file with a date and time stamp, or the exact filename of the file
- **Alternate Upload Folder -** Please specify an alternate portal upload directory. By default all files will be uploaded under a directory called DynamicForms_Uploads within your portal root directory (usually /portals/0/DynamicForms_Uploads/).
- **File Save Type** - specify how you would like the filename to be saved in the database, this can either be just the name of the file or it can be the full HTTP path to the file

This screenshot demonstrates the “File Attachment/Upload” element as seen by the end users.

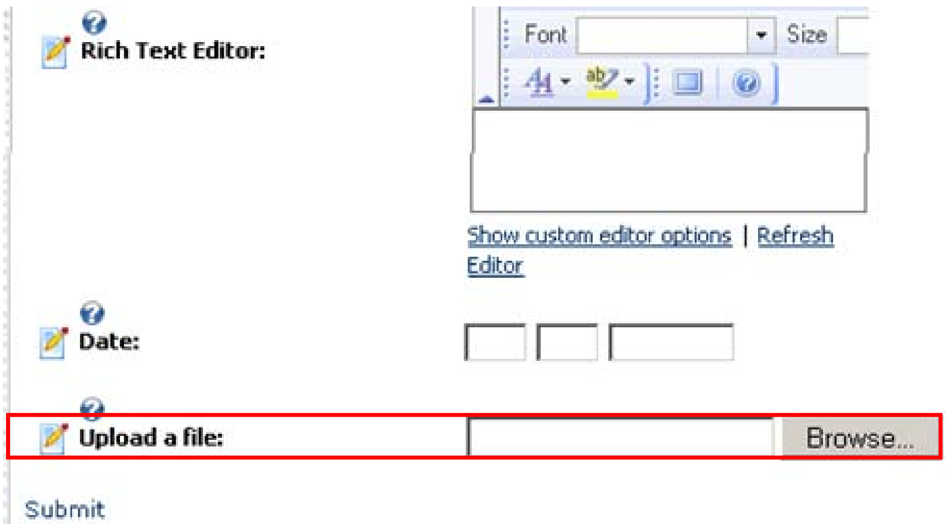


Figure 101: “File Attachment/Upload” element as seen by the end users

7.28 Creating a DNN® Country Element

The purpose of the “DNN® Country” element is to allow simple and quick setup of the pull down menu with the list of countries and add it to your dynamic form.

In order to start creating the country element, choose option “Country” inside the screen for creating a new dynamic question and click on the ”Update Field” link.

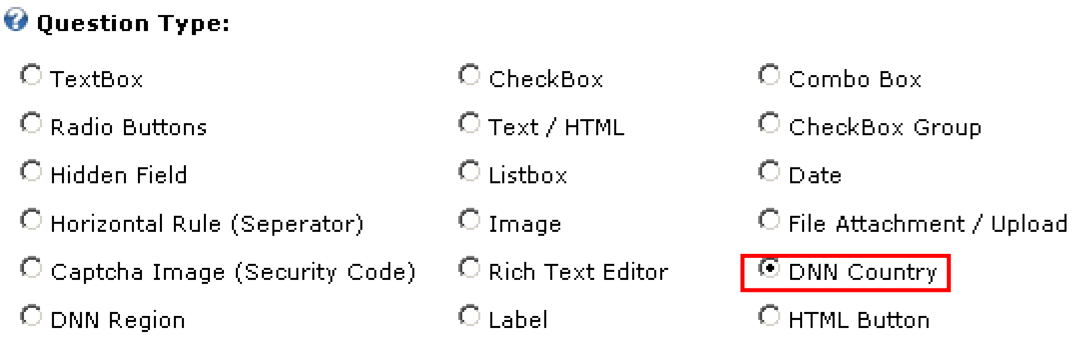


Figure 102: Creating a Country

The screenshot below demonstrates the country pull down menu as seen by the end users.

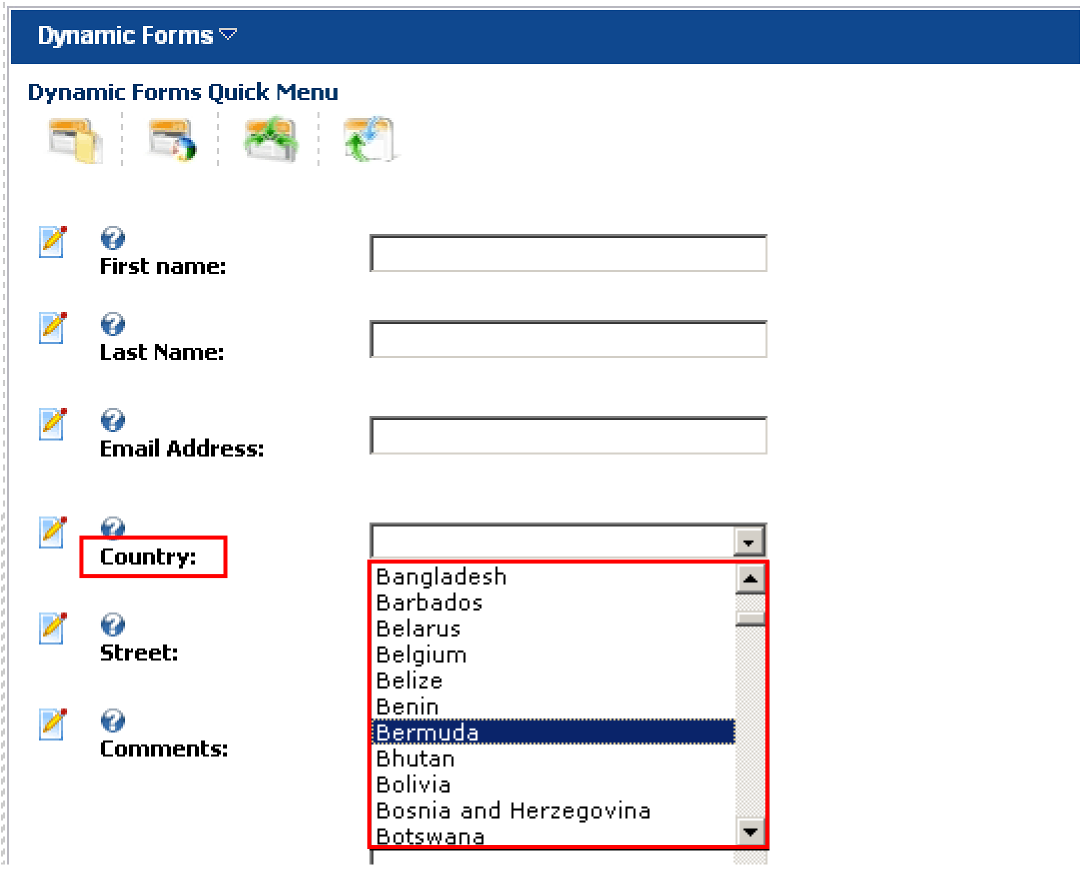


Figure 103: Example of the country element

7.29 Creating a Data Grid field

This option is used for creating a data grid field. In order to create this field, choose “Data Grid” when creating a new form item.

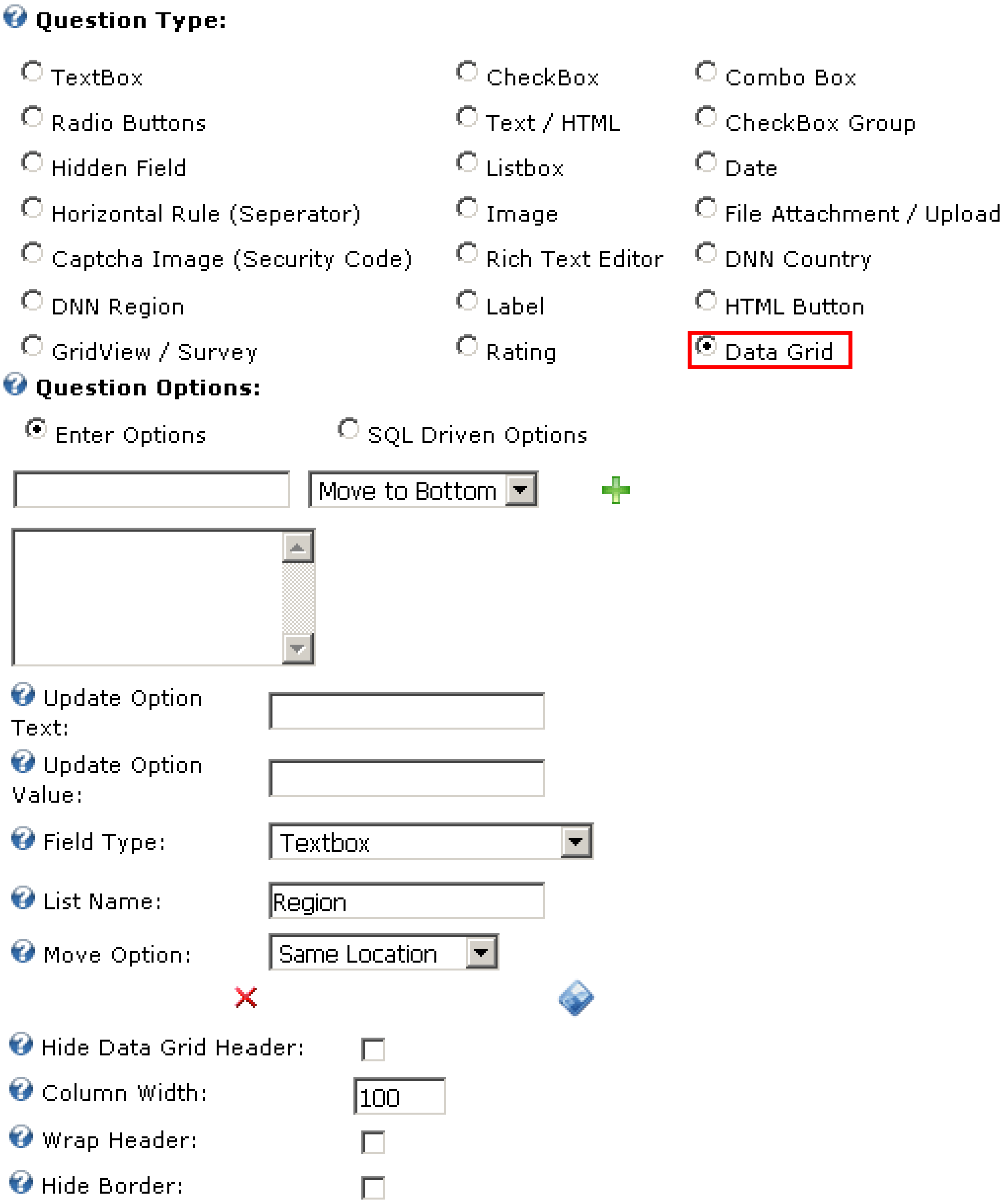
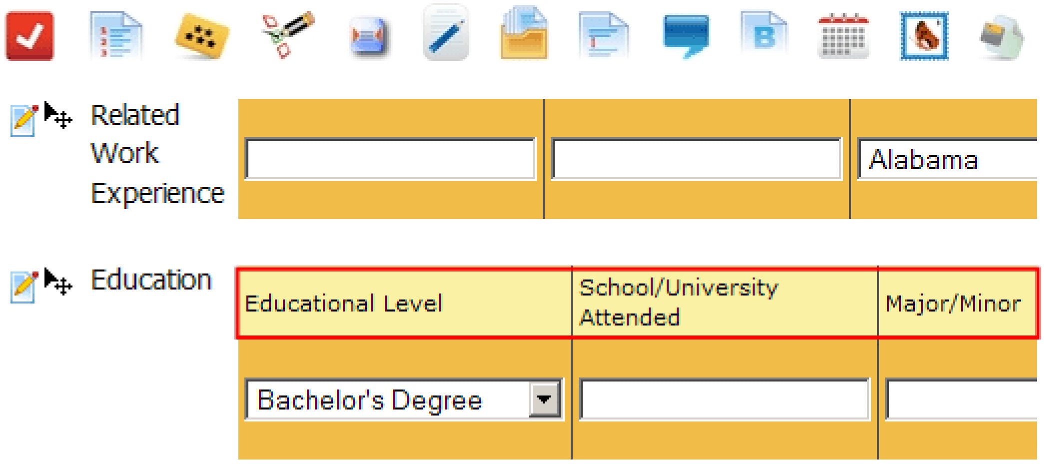


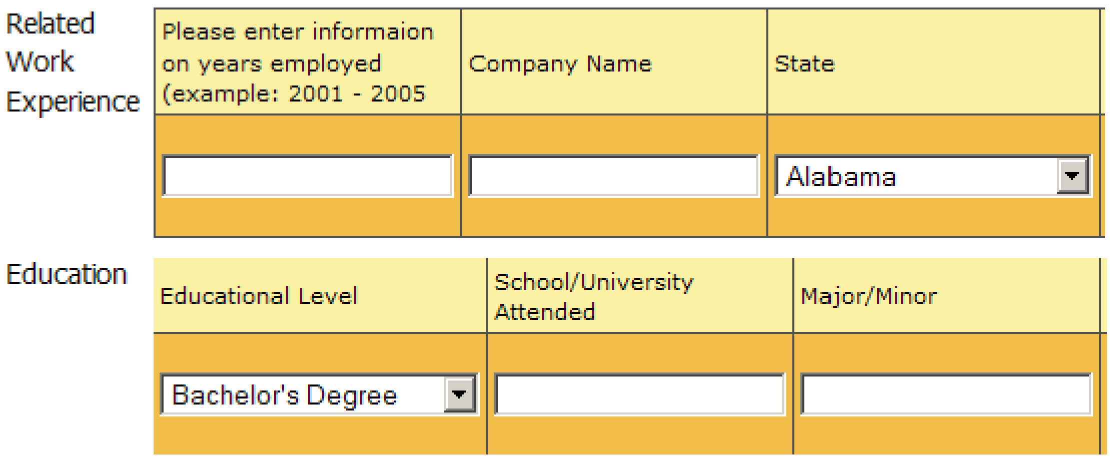
Figure 104: Creating a data grid field

The following “Data Grid” specific parameters are available:

- **Hide Data Grid Header -** select this option if you would like to hide the header for a data grid field (note the absence of the header for the first question “Related Work Experience” and the prominent header for the “Education”)



- **Wrap Header** – select this option if you would like to wrap the header for the data grid field. This feature will automatically wrap the text within the column/cell header if its larger than the cell width.
- **Hide border** - select this option if you would like to hide the border that the data grid normally displays (note the border around the "Related Work Experience" field and the absence of border for the "Education" field)



-

Note: see section 7.18 for further information on creating and managing options since the procedure is identical to creating a comb box.

7.30 Editing a question

In order to edit a question choose the “Manage Questions/Settings” option form the main menu and select the desired question from the pull down menu.

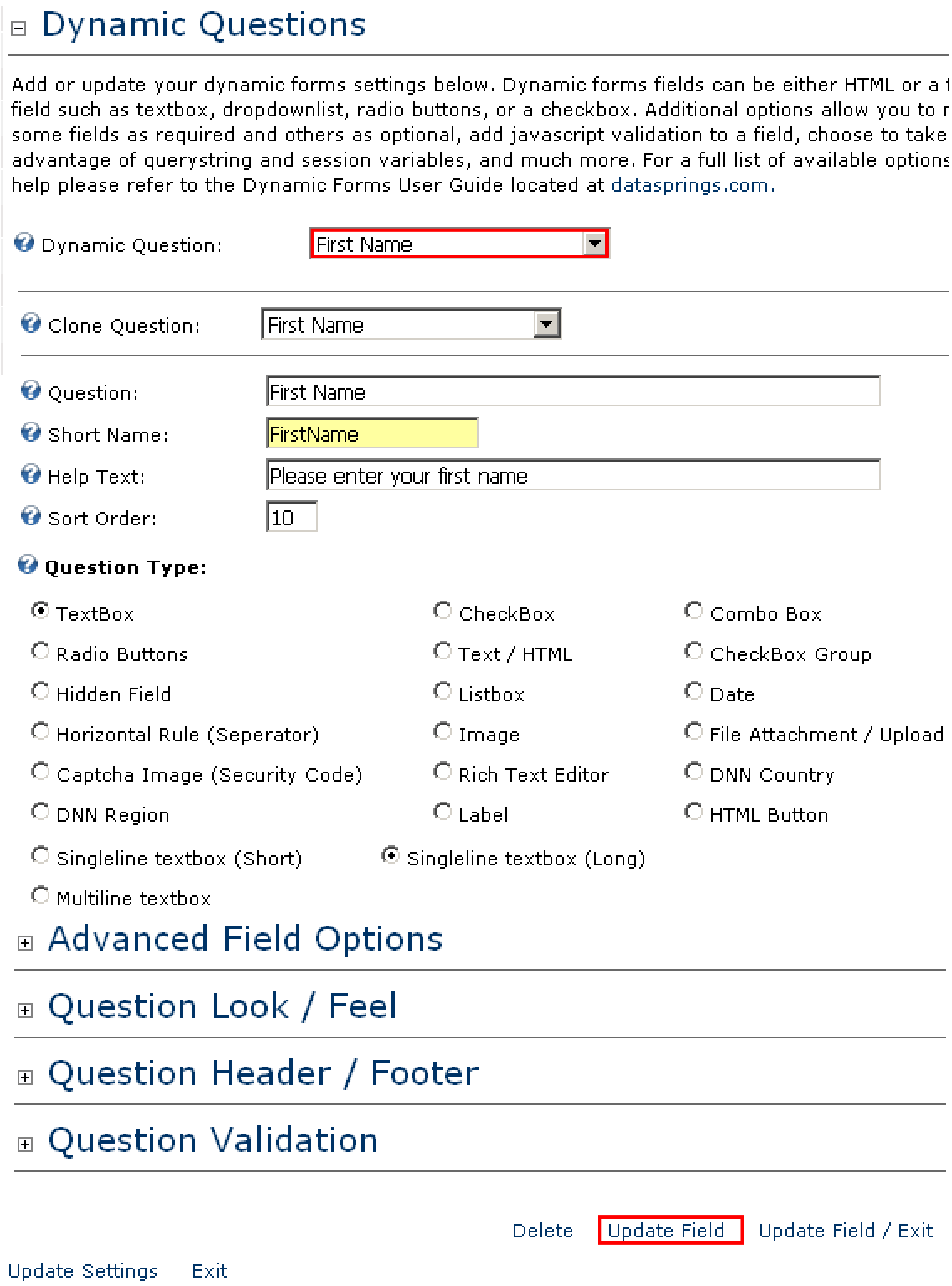


Figure 105: Editing a question

The screen with the parameters set for the chosen question will be displayed where you can make the desired changes and click on the “Update Field” in order to save them.

Note: see section 7.2 for further information about the question parameters.

7.31 Deleting a question

In order to delete a question, choose the desired question from the “Dynamic Question” pull down menu and click on the “Delete” link.

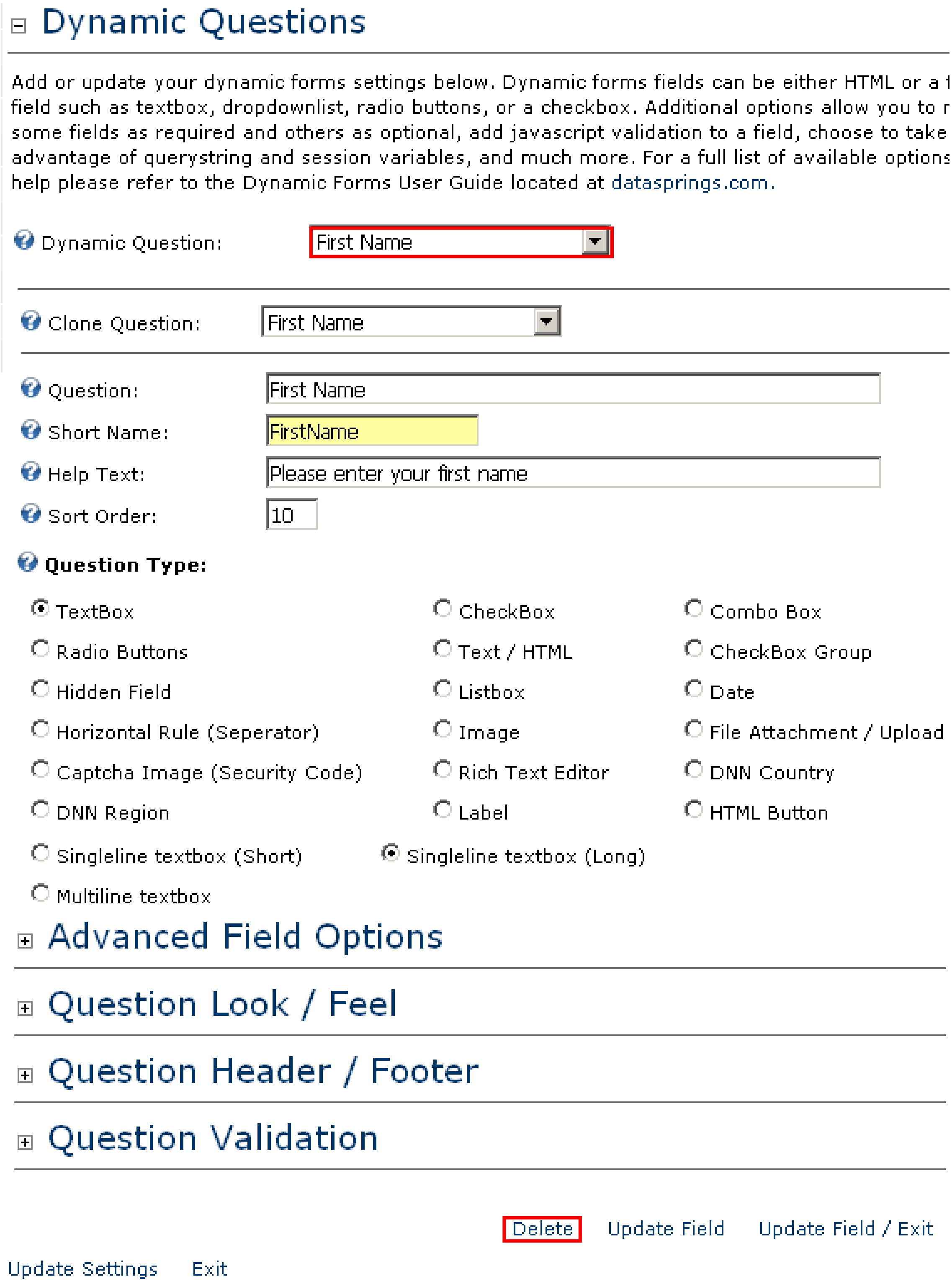


Figure 106: Deleting a question

Once you click on the “Delete” link, the selected question will be deleted.

7.32 Managing Module Configuration

In order to start managing general settings choose the "Module Configuration" option.

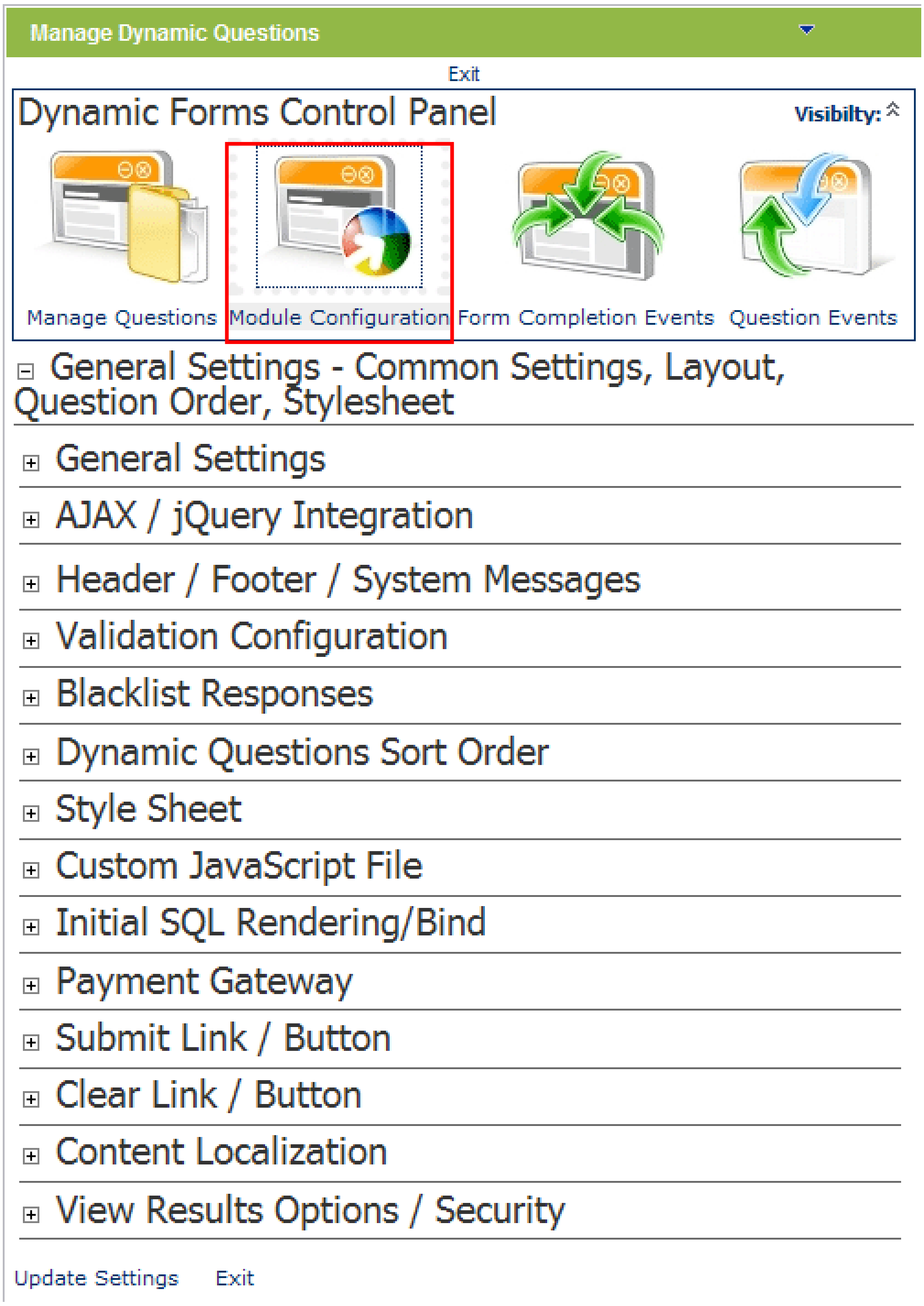


Figure 107: Managing general settings – Common Settings, Layout, Question Order, Stylesheet

The following options are available inside this screen:

- **General Settings** – see section 7.33
- **Header/Footer/System Messages** – see section 7.34
- **Validation Configuration** – see section 7.35
- **Blacklist Responses** – see section 7.36
- **Dynamic Questions Sort Order** – see section 7.37

- **Style Sheet** – see section 7.38
- **Custom JavaScript File** – see section 7.39
- **Initial SQL Rendering/Bind** – see section
- **Payment Gateway** – see section 7.42
- **Submit Link / Button** – see section 7.47
- **Clear Link / Button** - see section 7.48
- **Content Localization** – see section 7.49
- **View Results Options/Security** – see section 7.50

7.33 Managing General Settings

In order to start managing general settings click on the "+" symbol next to the "General Settings" label.

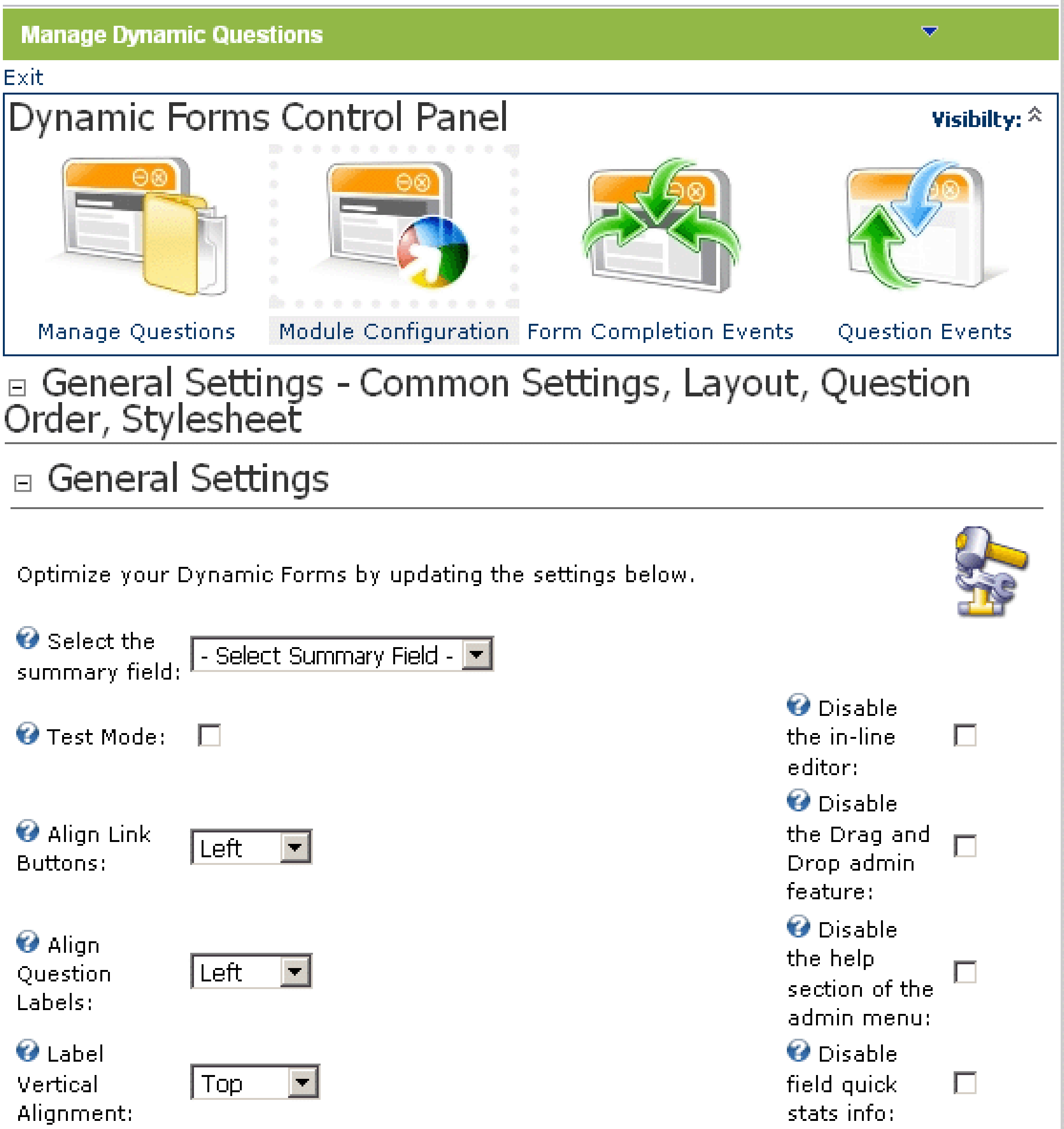


Figure 108: Managing General Settings

The following parameters are available inside this screen:

- **Select the summary field –** the summary field is the field you would like to be displayed within the form results grid.
- **Test Mode** – check this checkbox if you want the application to run in test mode
- **Align Link Buttons –** align the 'save/submit/complete' link button towards the left or right of the module.
- **Align Question Labels –** align the question labels towards the left of the label cell or right of label cell.
- **Label Vertical Alignment –** align the label vertically within the field rows either top, within the middle of the cell, or towards the bottom of the cell.
- **Field Vertical Alignment -** align the field vertically within the field rows. Options include either top, middle, or bottom of the cell.
- **Label Cell Width –** depending on how long your field labels are you might want to expand the width of the label cell. For example if you have a field label of: "Please provide your first name." you might want to have a larger width then if your field label is "First Name:".
- **Hide Help Icons** – select this checkbox in case you wish to hide the help icons
- **Only allow form submission once per user** – select this checkbox if you wish to prevent the users from using the form more than once.
    - **Note:** You can specify the message the user receives if they navigate back to the page to submit the form again in the 'Layout / System Messages Section'
- **Force fields labels and fields on same column?** – select this checkbox if you wish to force the fields labels and fields to appear on the same column
- **Enter key forces form submission** - select this checkbox if you would like the form to be submitted if the user clicks the 'Enter key' on the page
- **Set focus to first field on form?** - select this option if you would like the form to set the focus to the first field after loading
- **Pass unique completionID?** – select this checkbox if you wish to pass along a unique completionID session variable. This variable can then be picked up by 3rd party modules or other Data Springs modules such as 'Tailored Text/HTML' to retrieve form results
- **Display message to user after initial save?** – check this checkbox if you want to display the message to the user after initial save
- **Link text to continue** - link text displayed if you select to display a message after the form is submitted. I.e. "Thanks for submitting the form". **"Click Here" to continue…**
- **Default Short Field Length** – input field for setting default short textbox length i.e. maximum allowed number of characters
- **Default Long Field Length** - input field for setting default long textbox length i.e. maximum allowed number of characters
- **Question Suffix** – enter the suffix that will be appended to each question
- **Check blacklist for invalid responses** – select this checkbox if you want to enable checking blacklist for invalid responses
- **Word Wrap Field Label** – select this checkbox if you want to word wrap the field label
- **Redirect Page After Save** – choose the page user should be redirected to after clicking on save
- **Disable the in-line editor** – specify this feature if you wish to disable the in-line label and property editor. The in-line editor is the pencil icon that allows you to change field labels, question header/footer, form header/footer and other changes directly on the user-facing form directly without needing to go into the manage questions / settings area. To disable this feature you can check this box.
- **Disable the Drag and Drop admin feature** - select this checkbox if you wish to disable the drag and drop features within the administration user-facing form

- **Disable the help section of the admin menu** - select this checkbox if you wish to hide the help section of the administration user-facing form
- **Disable field quick stats info** – select this option to disable the quick stats field icon next to each field when in edit mode
- **Disable form AJAX** – select this option to disable AJAX within the form for fields and question events (**note**: this option is useful because sometimes some field types will not render properly or save properly if AJAX is enabled with a combination of some fields)
- **Disable jQuery** - select this option to disable the jQuery (**note:** some field types require jQuery such as the ratings control. If jQuery is enabled you can also utilize it for client side events
- **Disable AJAX within module configuration** - Disable AJAX features within the module configuration and control panel administration area
- **Disable editing results** – select this option to disable the editing form results feature within the module. The editing results is a token that can be used within email events and also within the module 'View Results' section.

The screenshot below demonstrates the layout of the form when the labels and the fields have not been forced on the same column.

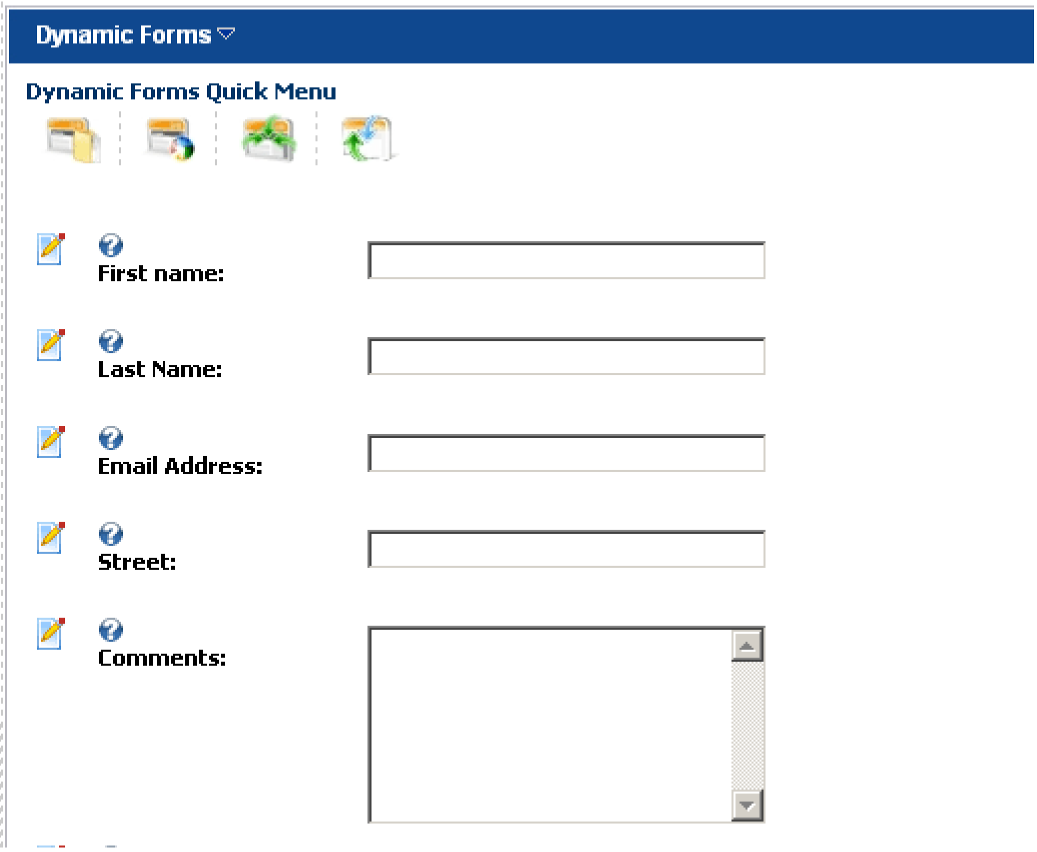


Figure 109: Normal layout of the fields and labels

In order to force the fields and labels on the same column, select the "Force fields labels and fields on same column?:" checkbox within the "General Settings" page.

The screenshot below demonstrates the layout of the form in case the fields labels and fields have been forced to appear on the same column.

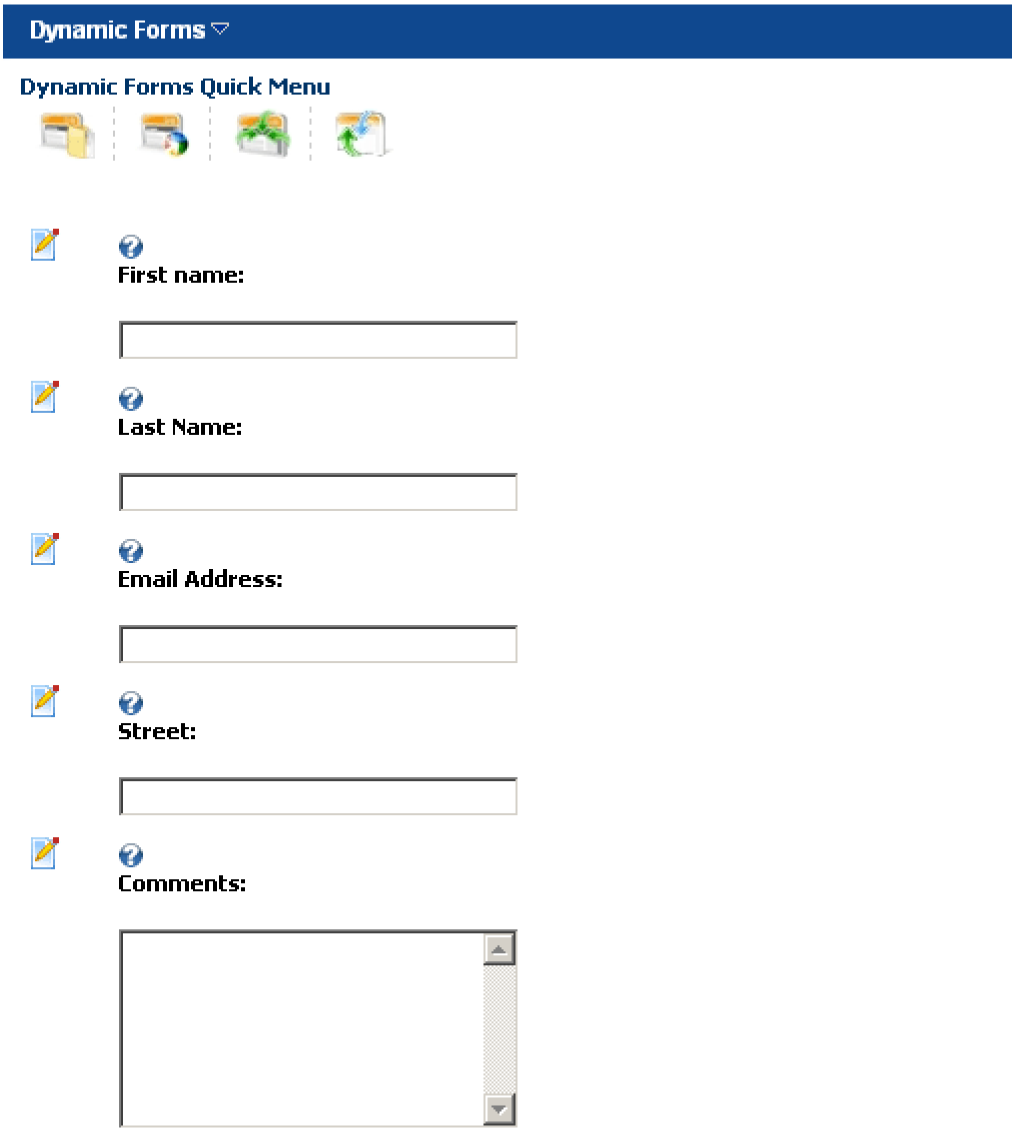


Figure 110: Example of the form in case the fields and labels are forced on same column

7.34 Managing Header/Footer/System Messages

In order to start managing header, footer or system messages, click on the “+” symbol next to that label.

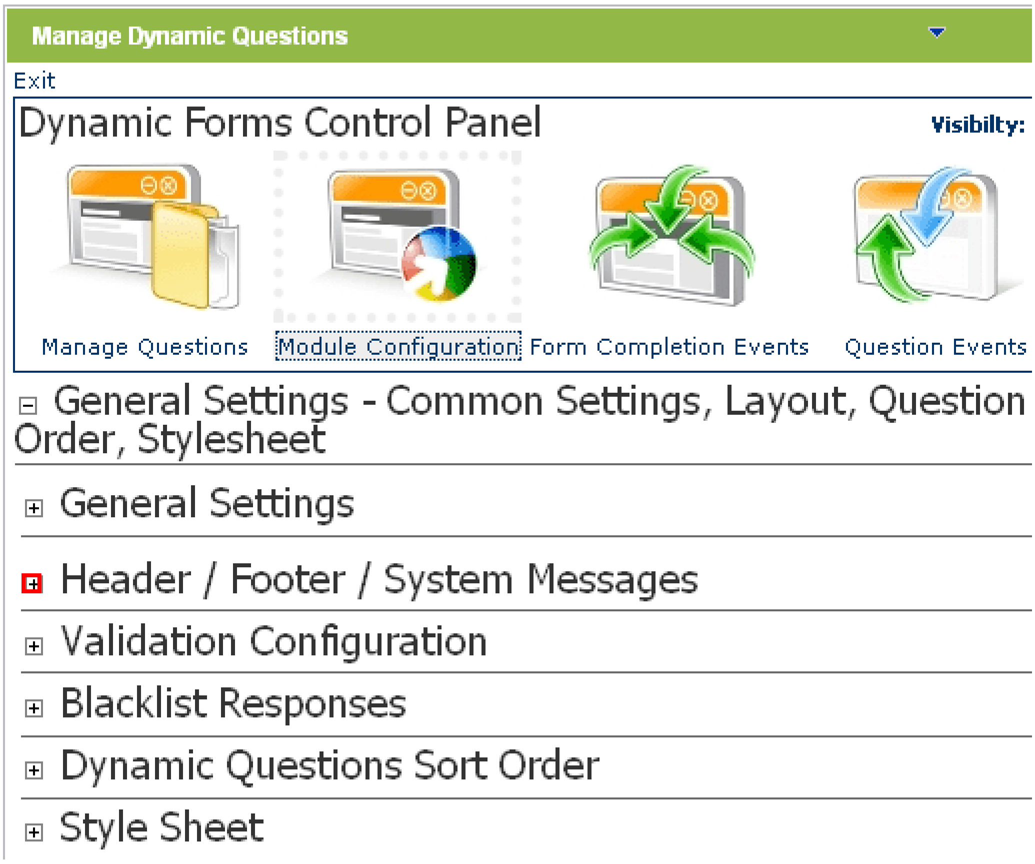


Figure 111: Choosing option “Header/footer/System Messages”

The following screen will be displayed.

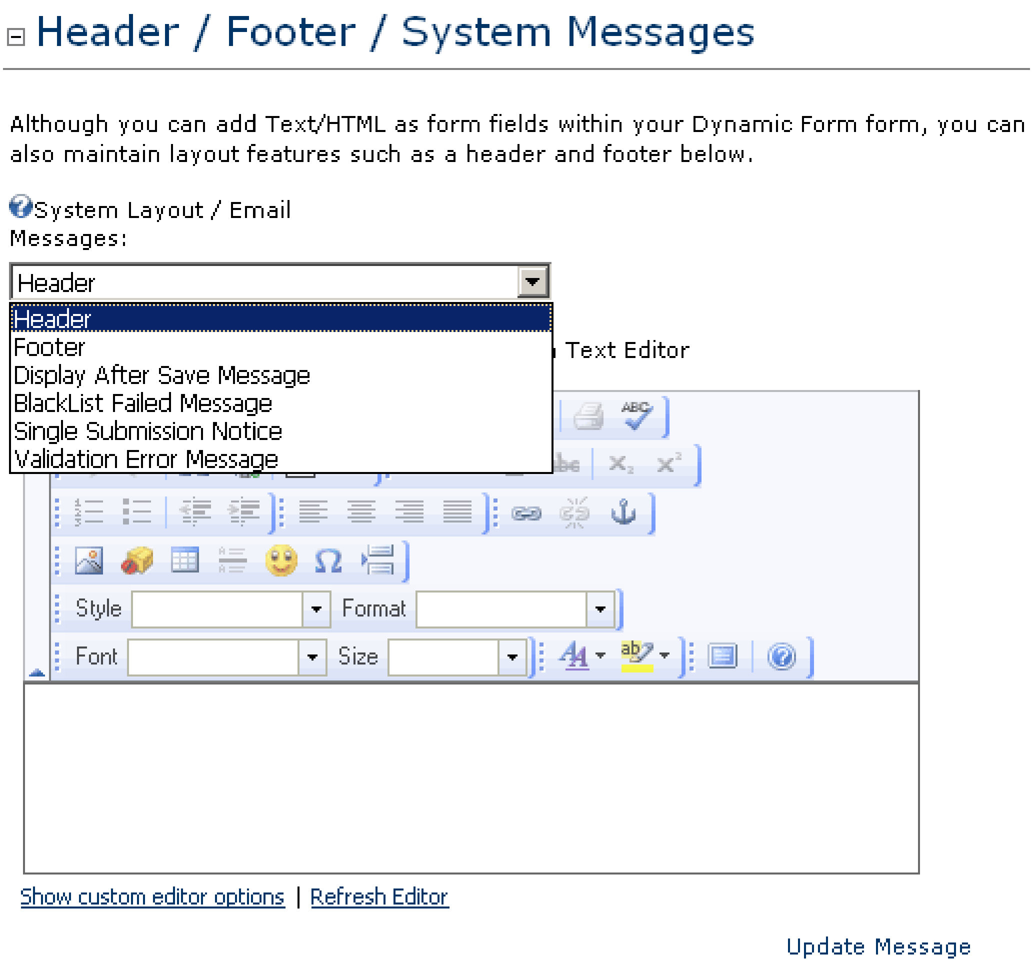


Figure 112: Managing header/footer/system messages

- **Header** – select this option in order to define the introductory text for your form (**note**: displayed as a header above the form)
- **Footer** – select this option if you want to enter any additional information for the user after they have completed the form (**note**: displayed as footer below the form)
- **Display After Save Message** – option for defining a message that will be displayed to the user after filling the form and submitting information (**e.g** "thank you for submitting information")
- **BlackList Failed Message** – option for defining a message which will be displayed to a user posting an unacceptable response within the form (**e.g.** "This kind of response is considered unacceptable") – see section 7.36.1 for further information on creating blacklist responses
- **Single Submission Notice** – use this option to define the message displayed to the user in case they have tried to submit the form more than once (**note**: this option is used in case you wish to allow the users to submit the form once only; for this to work you should also select the "**Only allow form submission once per user"** option within the settings page - see section 7.33)
- **Validation Error Message** - use this option to define the message displayed to the user in case there has been a validation error

After setting the desired parameters, click on the "Update Message" link in order to save the changes.

7.35 Managing Validation Configuration

Dynamic Forms allows you to manage various options related to the way the responses from your users are being validated.

In order to enable start configuring the validation methods, click on the "+" symbol next to the "Validation Configuration" label (after clicking on "Module Configuration" within the "Settings" page).

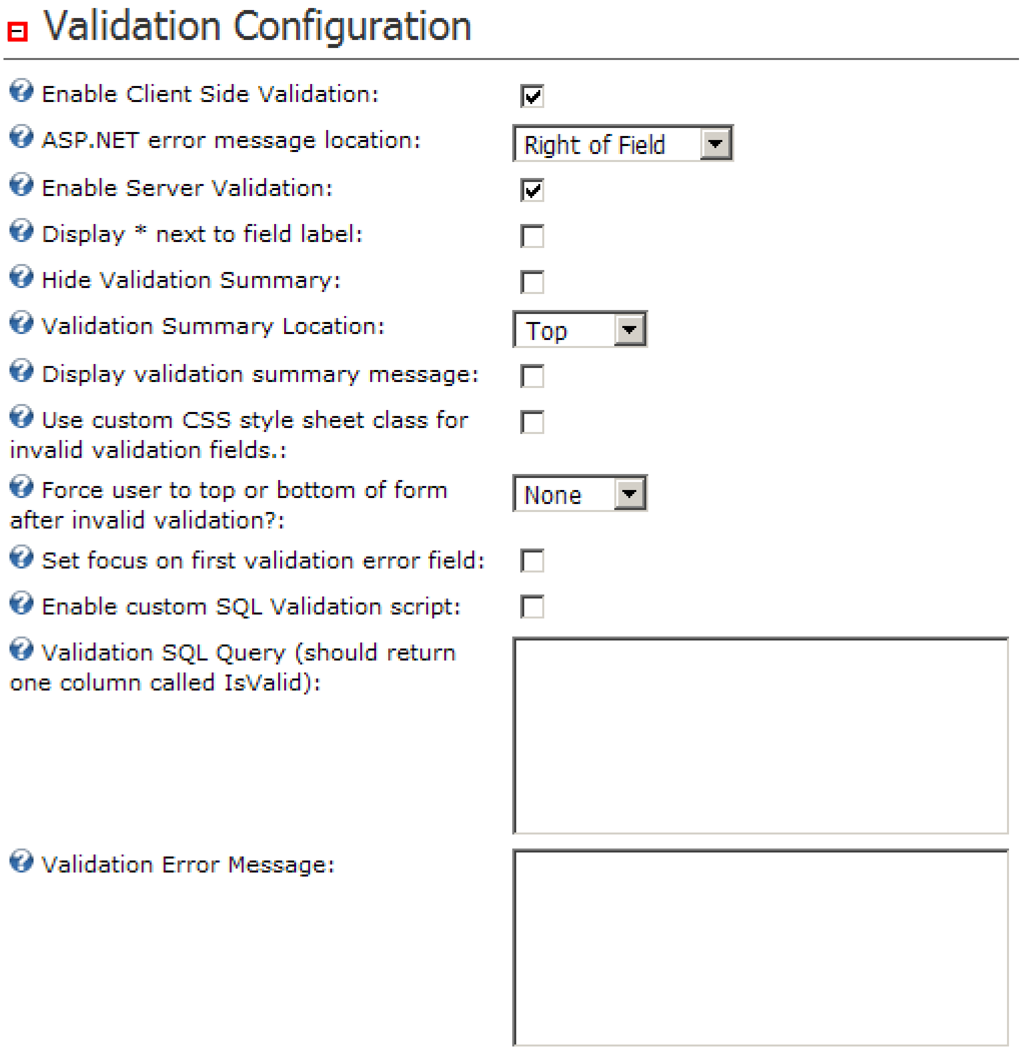


Figure 113: Managing Validation Configuration

The following parameters are available within this screen:

- **Enable Client Side Validation** – select this checkbox to enable the client side validation (**Question:** the validation will be performed via JS/Ajax?) – (note: this feature will enable ASP.NET Client Side Validation controls; these include regular expression validations, compare validations, and standard validation for fields which will display a * - or expression text - directly next to the field.
- **ASP.NET error message location** - select the location of the error messages which appear next to the ASP.NET Client Side validation controls (these are usually displayed as * next or below the field)
- **Enable Server Validation** – select this checkbox to enable the server side validation. Server side validation functions separate from client side validation and does not use ASP.NET validation controls. Server side validation is required for certain field types such as checkbox, Captcha, Country, Region, Image, File Upload, and multi selection field types such as Check Box Lists.
- **Display * next to field label** - select this option if you would like an asterisk (*) to appear to the right of each field label

- **Hide Validation Summary** – select this checkbox if you wish to hide the validation summary. The validation summary is a bulleted list of invalid form fields and is separate from an text that appears directly next to fields. This summary (when visible) is either at the top or bottom of the form depending on the validation summary location setting.
- **Validation Summary Location** – use this pull down menu to set the desired location of the validation summary (**e.g.** “Top”)
- **Display validation summary message** – select this option to display the validation summary message. The validation summary message can be defined in the ‘Header / Footer / System Messages are. The setting to modify this message is ‘Validation Error Message’ within the dropdownlist (see section 7.34)
- **Use custom CSS stlylesheet class for invalid validation fields** – select this checkbox in case you wish to use custom CSS Stylesheet class for invalid validation fields. This feature will change the CSS class for fields that have been flagged as invalid during the validation process. (*server validation only*)
    - ***Note**:* The CSS class for these invalid form fields will be set to: DynamicForms_FieldError
        - ***Example:***

          .DynamicForms_FieldError

          {

          background-color: red;

          }
    - The CSS class can be setup within the styles sheet section of module configuration (see section 7.38)

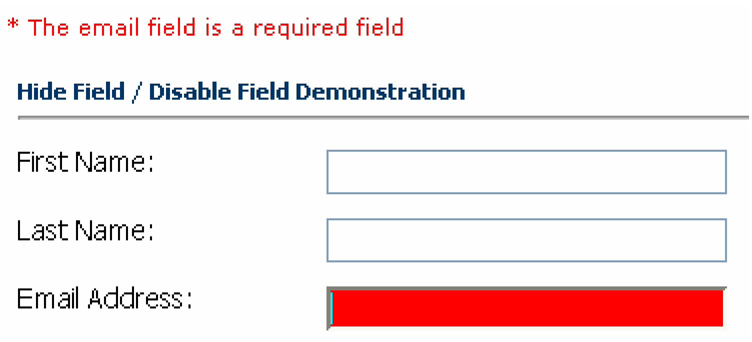


Figure 114: Example of the CSS setting applied to a field

- **Force user to top or bottom of form after invalid validation** – choose the desired location for forcing the user after a response has been marked as invalid (*server validation only*)
- **Set focus on first validation error field** – select this option if you wish the page to jump to the location of the first spotted error in user response i.e. move to that exact location within the page (*server validation only*)
- **Enable custom SQL Validation script** – select this option you would like to enable a custom SQL Validation Script (**<u>note:</u>** validation via SQL should *always* use stored procedures to prevent SQL Injection when referencing fields)
- **Validation SQL Query (should return one column called IsValid)** - enter an SQL query to perform custom validation for this field; the SQL query should return one column called **IsValid**; If the column returns 0 or False then the validation error message will be thrown, anything else and the user can continue; <u>validation via SQL should *always* use stored procedures to prevent SQL Injection when referencing fields</u>
- **Validation Error Message** – enter the text which will be displayed to the user in case a validation error occurred

7.36 Managing Blacklist Responses

Dynamic Forms allows you to create a 'blacklist' of unacceptable values or responses. After the user submits such a response, he will be notified that the response was inappropriate and the response will not be saved.

In order to enable the “Blacklist” feature you must enable the setting “Check blacklist for invalid responses” under “Module Configuration” (see section 7.32).

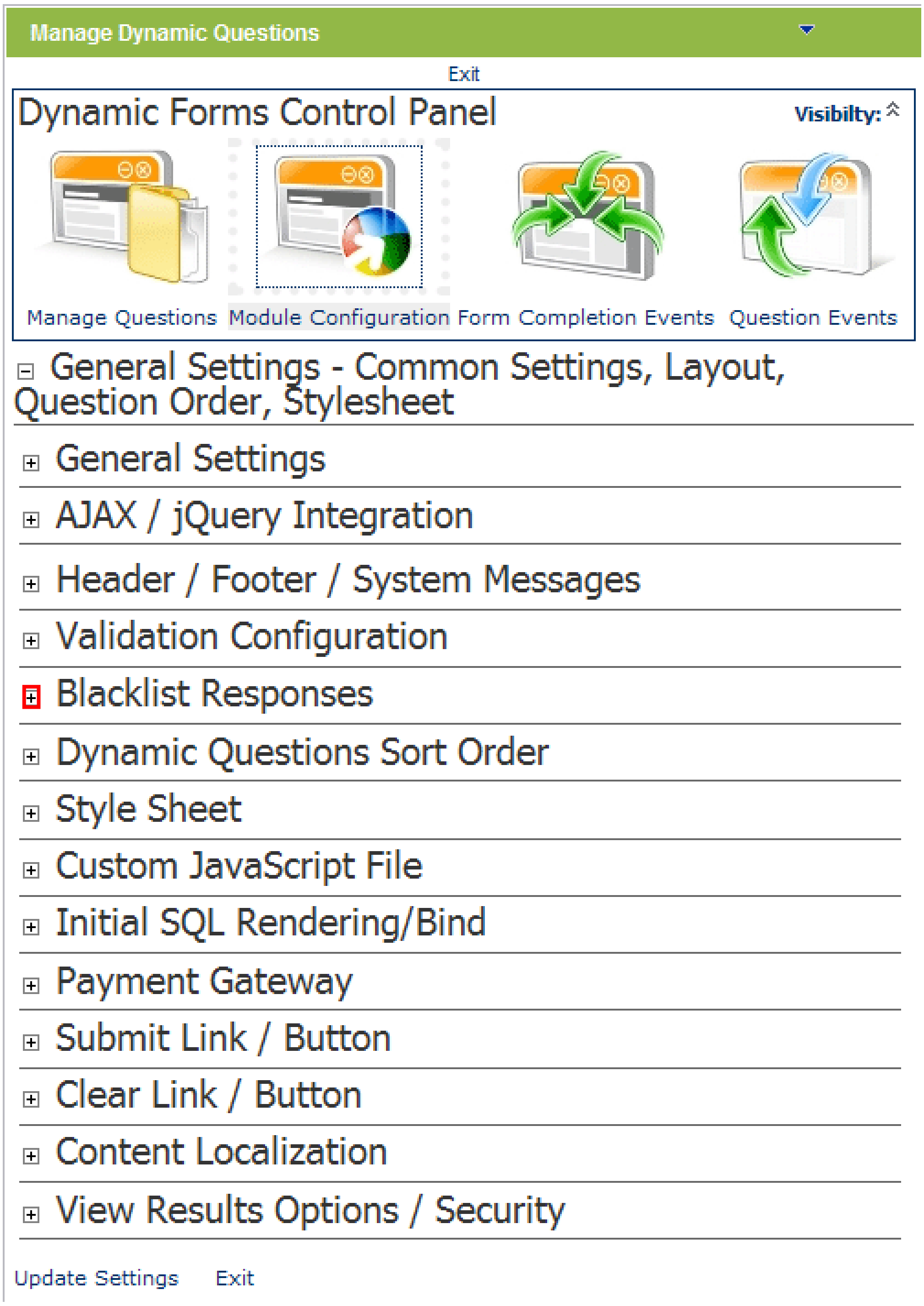


Figure 115: Choosing option "Blacklist Responses"

The following screen will be displayed.

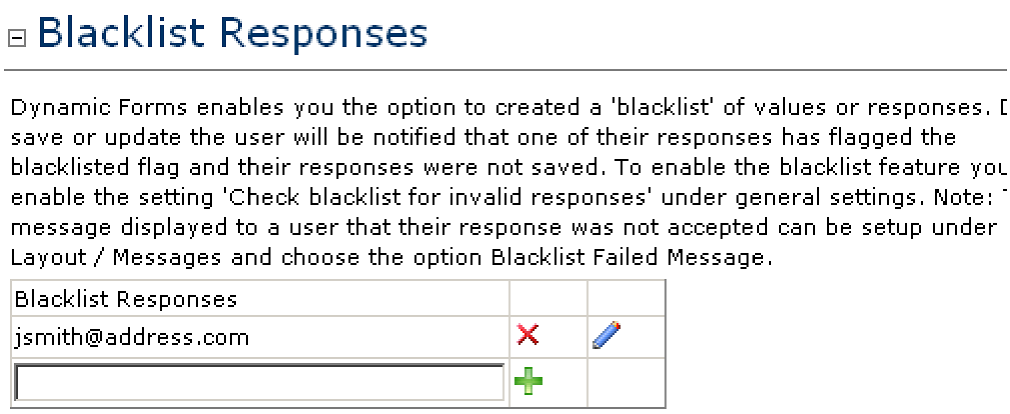


Figure 116: Managing black list responses

The following options are available inside this screen:

- - option for creating a new blacklist response (see section **7.36.1**)
- - option for editing a blacklist response (see section **7.36.2**)
- - option for deleting a blacklist response (see section **7.36.3**)

7.36.1 Creating a Blacklist response

In order to create a blacklist response, enter the desired response into the input field and click on the add icon .

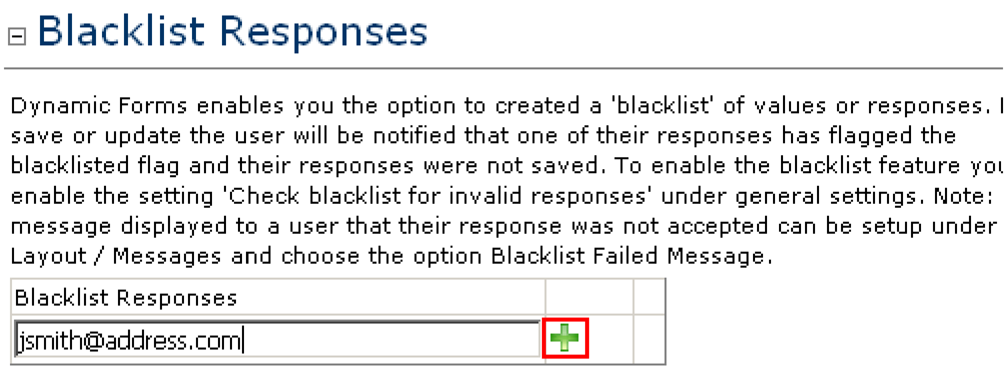


Figure 117: Creating a blacklist response

The new blacklist response will be created.

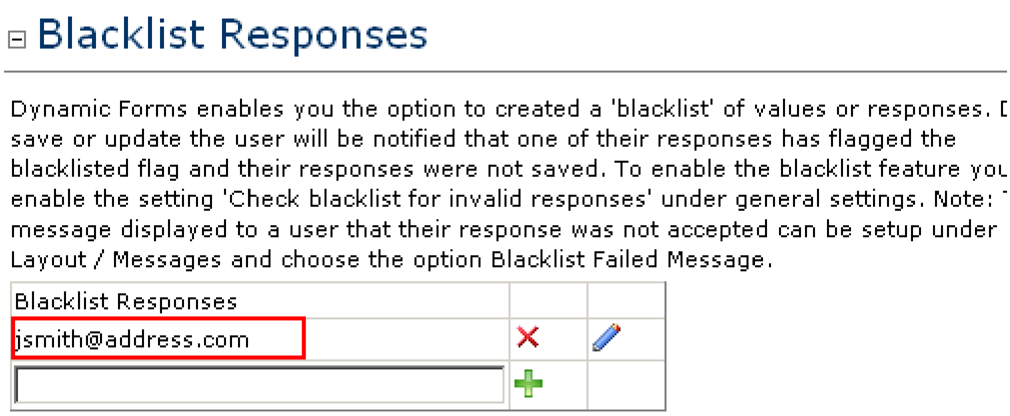


Figure 118: Blacklist response created

From this point on, this response will be treated as unacceptable and users posting such a response will be notified of this fact.

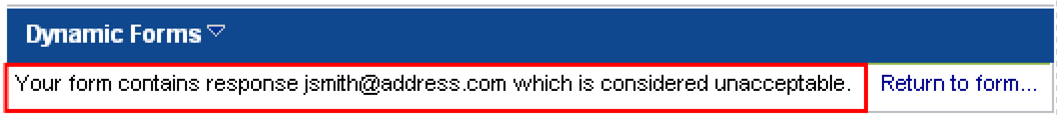


Figure 119: Example of the message displayed to the user posting blacklist response

Note: the error message seen in the screenshot can be set under the “Layout Settings” section.

7.36.2 Editing a blacklist response

In order to edit a blacklist response, click on the edit icon next to the desired response.

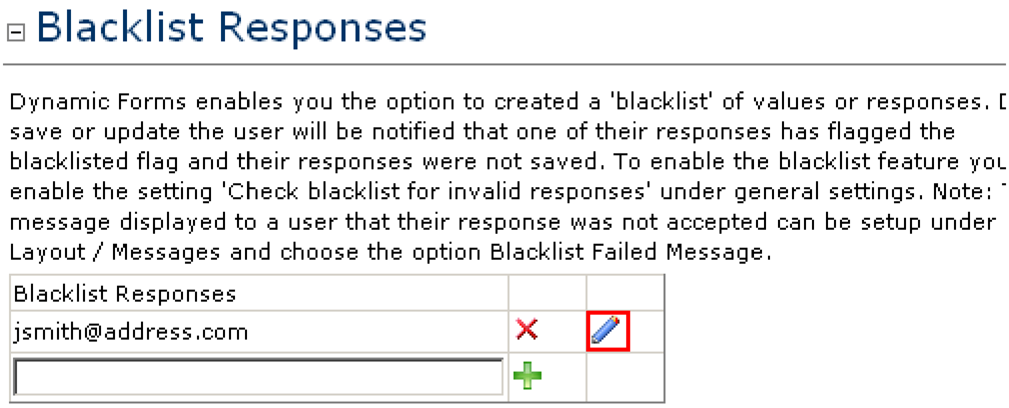


Figure 120: Editing a blacklist response

Change the response and click on the “Update Settings” in order to save the changes.

7.36.3 Deleting a blacklist response

In order to delete a blacklist response, click on the delete icon next to the desired response.

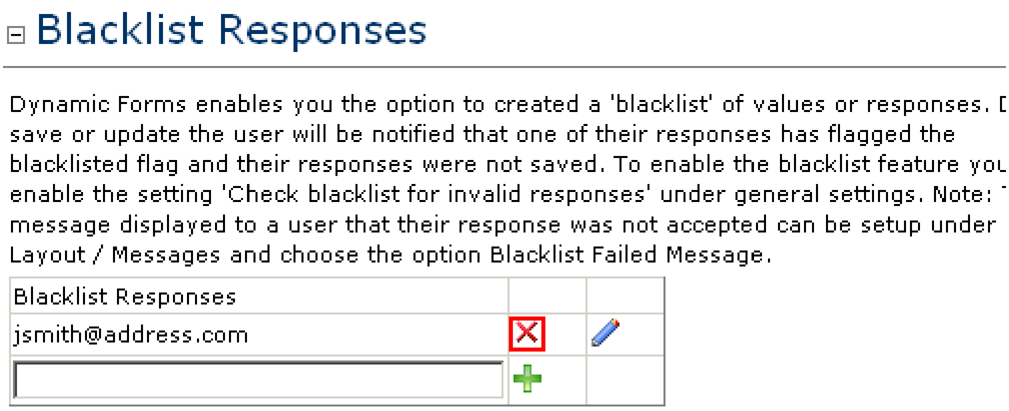


Figure 121: Deleting a blacklist response

Once you click on the delete icon, the selected blacklist response will be deleted.

7.37 Setting the Dynamic Questions Sort Order

In order to set the desired sort order for the dynamic questions, choose option “Dynamic Questions Sort Order” after clicking on the “Module Configuration” option.

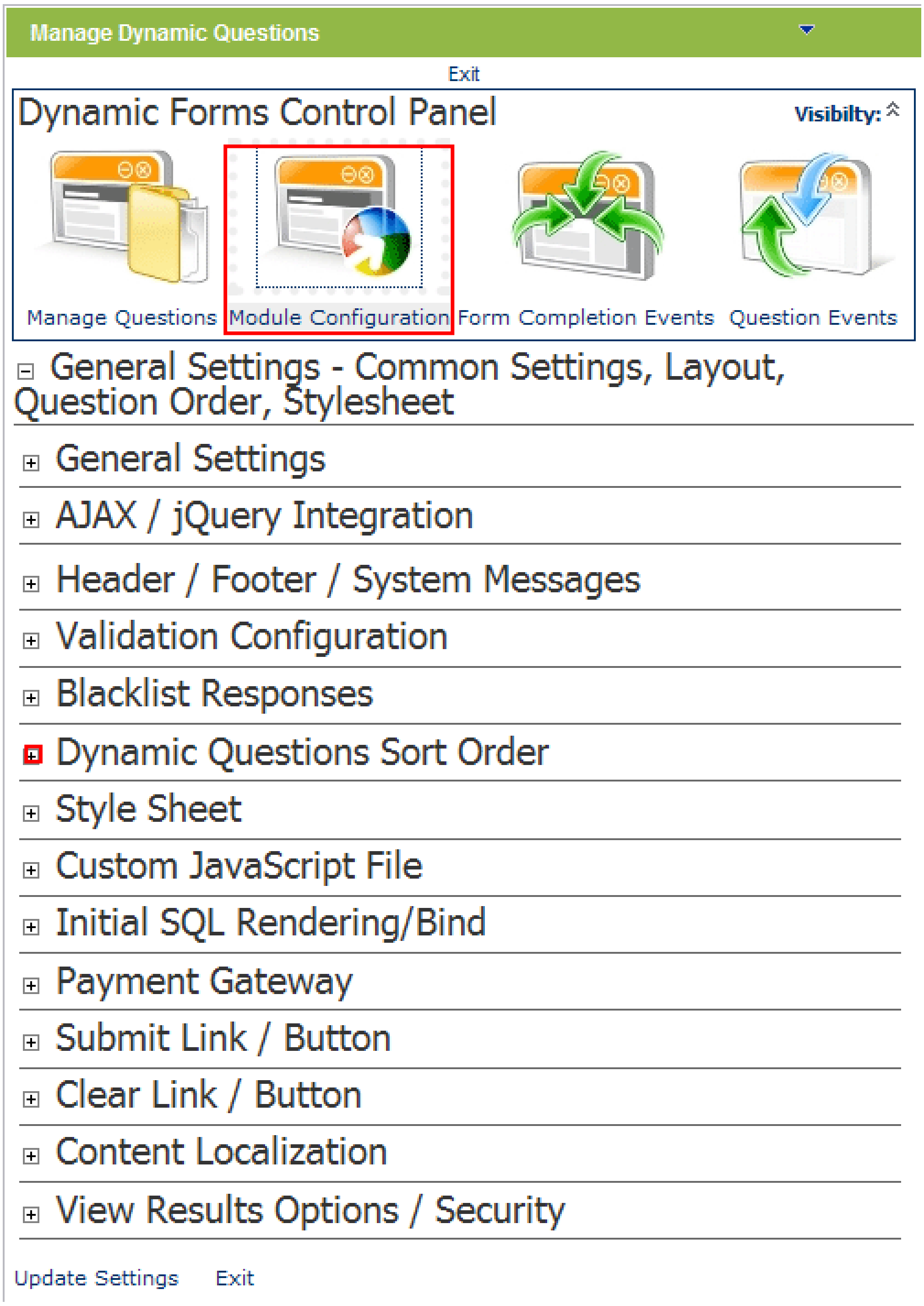


Figure 122: Setting the Dynamic Questions Sort Order (step 1/2)

The following screen will be displayed.

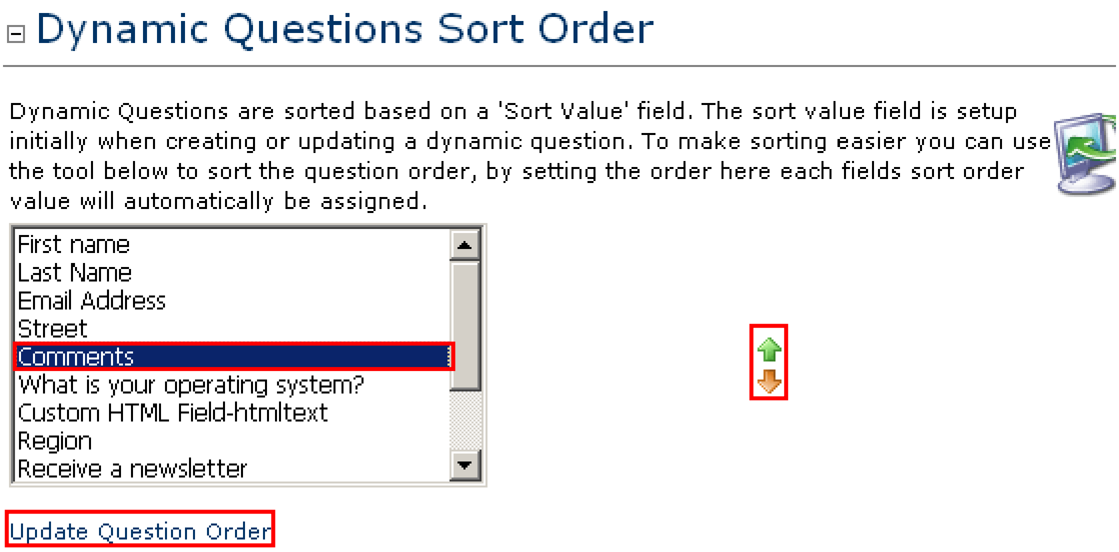


Figure 123: Setting the Dynamic Questions Sort Order (step 2/2)

Select the desired question and use the up and down arrows to set the desired sort order. Once you set the desired sort order, click on "Update Question Order" link to save the changes.

Note: With v2.3 Question Sort order is now done via ASP.NET AJAX / JavaScript. This helps with improved performance and quicker sorting as no postback occurs to the server until you click 'Update Question Order'.

7.38 Modifying the Style Sheet

In order to modify the style sheet, choose option “Style Sheet” after clicking on the “Module Configuration” option.

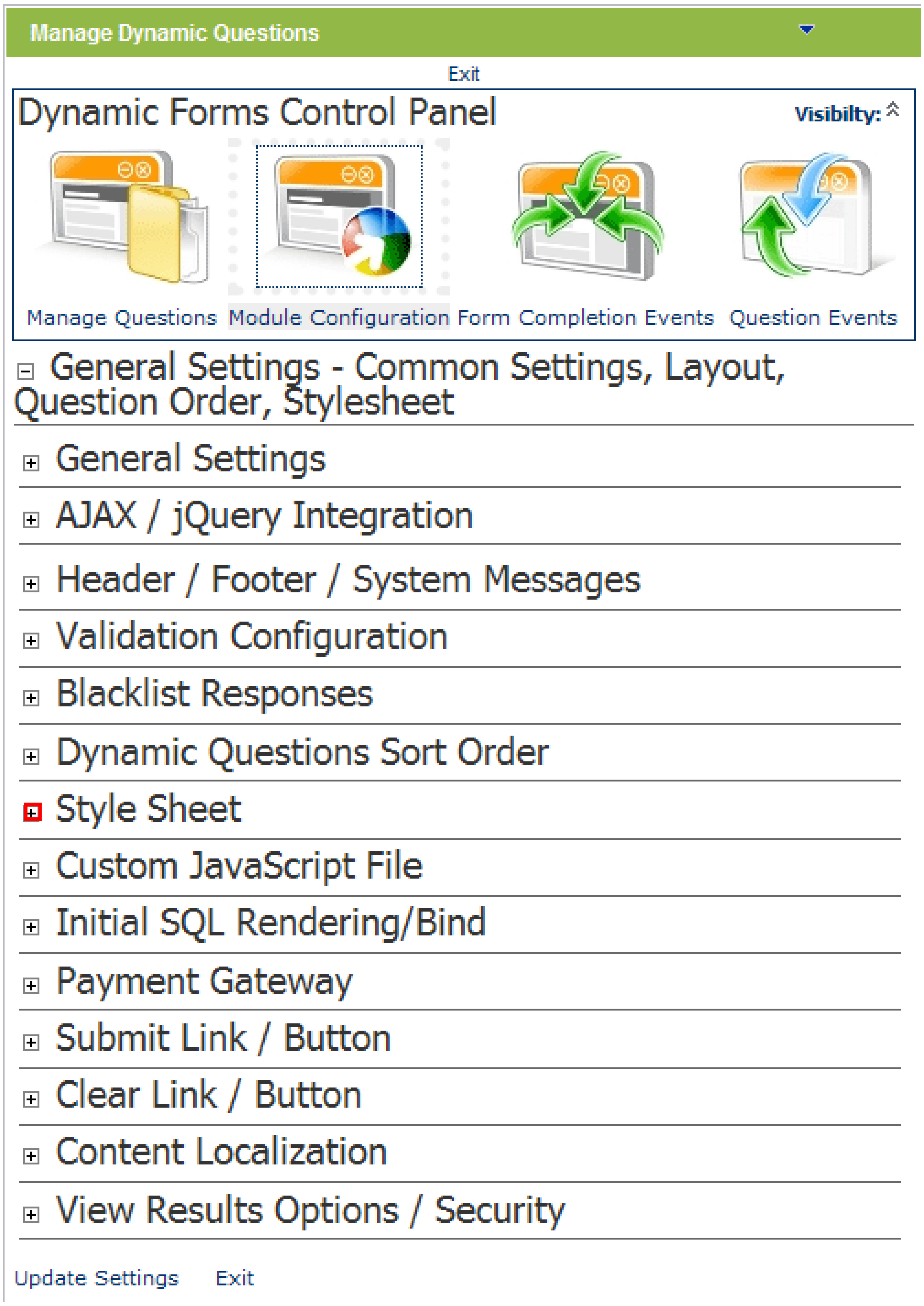


Figure 124: Modifying the Style Sheet (step 1/2)

The following screen will be displayed.

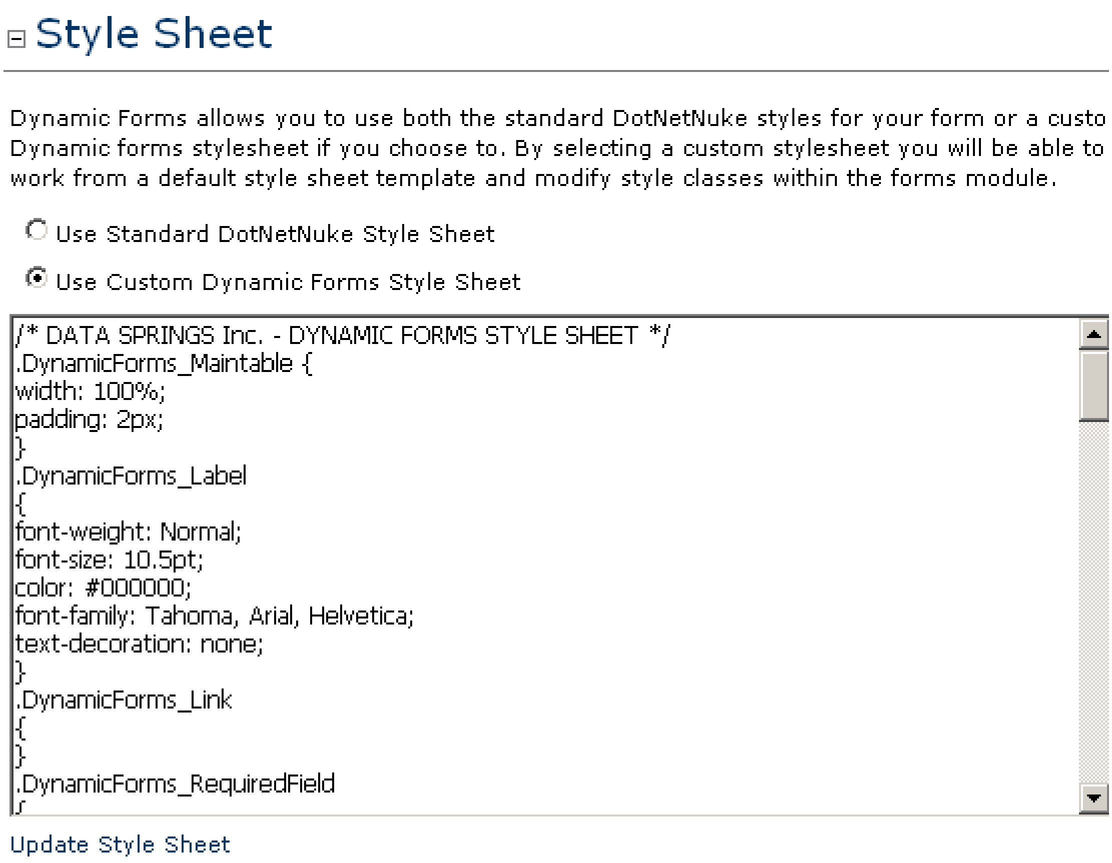


Figure 125: Modifying the Style Sheet (step 2/2)

The following parameters are available inside this screen:

- **Use Standard DotNetNuke® Style Sheet** – leave this option if you want to use the standard style sheet
- **Use Custom Dynamic Forms Style Sheet** – choose this option in order to enable the text area containing style sheet tags where you can modify the desired parameters

After setting the desired parameters, click on the "Update Style Sheet" to save the changes.

Note: After updating the style sheet and exiting the settings area, you MUST refresh the browser before the new style sheet settings will take effect. You can refresh the page by hitting F5 within the browser.

7.39 Managing Custom JavaScript File

In order to start managing the custom JavaScript file, choose option “Custom JavaScript File” after clicking on the “Module Configuration” option.

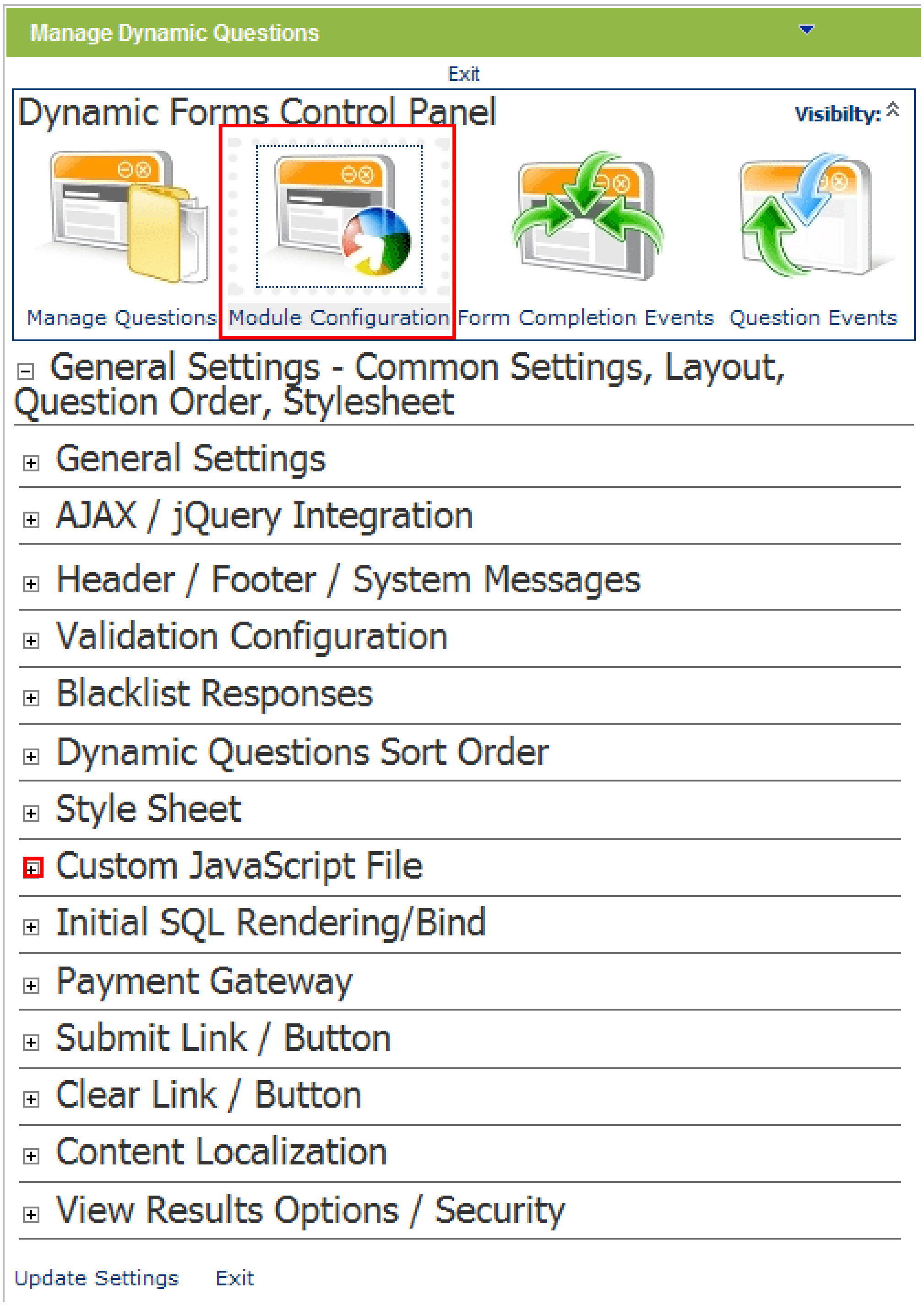


Figure 126: Managing the custom JavaScript file

The following screen will be displayed.

Figure 127: Managing the custom JavaScript file

The following options are available:

- **Initial Javascript** – use this text area to enter the initial javascript that will be executed when the initial form loads or when a post back occurs.
- **Submit/Save Javascript** – use this text area to enter the javascript that will be executed when the submission button is clicked.
- **Javascript file** - enter any custom javascript functions into the javascript file for this Dynamic Forms instance

7.40 Examples of Client Side Events

The JavaScript file is used for advanced settings as an option to enable any client side script you would like rendered when the form is rendered. A default file is included for scripts to calculate checkbox groups, create pop up help text, create pop up alerts for require fields, and other useful functions.

You can add/change any of the client side scripting code here so that you can enable additional client side code for your form. Please visit the Data Springs Product Forums for additional tips and tricks for client side code.

Note: The custom JavaScript file can be used along with the 'Client Side Event' feature for each form feature (please refer to section 7.3, setting up advanced field options)

Example client side events can include features to hide/show html, hide/show the form submission ability, pre-populate fields, calcualte checkbox fields, etc, You can review the following demonstration for more detailed tutorial on setting up some client side events:

Dynamic Forms Client Event Demonstration

Examples:

- **To fill the text of a field based on the text of another field**. This is assuming the short field names for the fields were DisplayName, FirstName, and LastName

  $(DisplayName) = $(FirstName) + ' ' + $(LastName)

- **To calculate check box group** (View demonstration). This is assuming the client event is placed on a checkbox group field with a short field name of 'Products' and that you are wanting to display the total in an HTML DIV tag called 'Total'.

  Total.innerHTML = '<B>Your total comes to: ' + '$' + CalculateCheckBoxListValues($(Products_FieldID), $(Products_ValueFieldID)) + ''

- **Hide form submission button unless the user types in 'YES' within a textbox**. This is assuming that the short field name for this field is Termsonkeyup and that you have enabled the feature within the link / submit buttons to 'Initially hide form submission button'.

  ```
  if ( $(Termsonkeyup).toUpperCase() == 'YES')
  {
  showhtml('SaveForm')
  }
  else
  {
  hidehtml('SaveForm')
  }
  ```

- **Hide form submission button unless the user checks a checkbox** (i.e. terms and conditions). This is assuming the client side event is applied to a checkbox field that has a short field name of 'chkShowHide'.

  ```
  if ( $(chkShowHide))
  showhtml('SaveForm')
  else
  hidehtml('SaveForm')
  ```

Client Side Events are processed based on fields onblur events (except where noted below). For textbox fields, you might want to process client side events in situations besides onblur. For these, you can use the following keywords within the fields short field name.

- **onclick** – Adds the client side event to the textbox onclick event

- **onfocus** – Adds the client side event to the textbox onfocus event
- **onkeypress -** Adds the client side event to the textbox onkeypress event
- **onkeydown -** Adds the client side event to the textbox onkeydown event
- **onkeyup** – Adds the client side event to the textbox onkeyup event

Would you like to to enable a client side event that you do not see an example of? Please write a post in the Dynamic Forms Product Forums area for community involvement. If we find the request useful we will create an example or tutorial and include the example in the next user guide.

7.41 Setting up the Initial SQL Rendering/Bind

The initial SQL rendering/ databind functionality allows you to build more interactive form implementations.

For example, a form that might be able to offer something such as an insert, edit, delete type functionality with tables within your database. As with all SQL functionality, you should always use caution to avoid SQL injection and we strongly recommend using a stored procedure to retrieve your results.

The initial SQL databind works similar to single field default value generation by SQL, however it can work for all fields with just this one query. The query should return a single data row and each column within the datarow that would like to retrieve should match a short field name within your query.

For example, 'Select DateOfBirth from YourTable where ID = 1'. If you had a field on the form with a short field name of DateOfBirth and this table returned a value, it would render the value within the form load.

In order to start managing the custom Initial SQL rendering/bind, choose option “Initial SQL Rendering/Bind” after clicking on the “Module Configuration” option.

[Dynamic Forms Initial SQL Rendering Demonstration]

Figure 128: Setting up the Initial SQL Rendering/Bind

The following options and parameters are available:

- **Enable initial SQL data bind** – select this checkbox to enable the form to initially render / bind data from an SQL query; the form will retrieve data based on the SQL table rendered and each fields short field name
- **Only enable when querystring value is present?** - enter the querystring parameter you would like to enable the initial SQL database.
    - **For example:** if you passed a querystring variable such as DBBind and a value of true it would enable the SQL binding feature, if it was not true it would not enable the binding feature. This is useful if you are wanting the binding feature to not be enabled for initial submission, however maybe you want to enable the binding feature to update a record. This feature only works if you already have enabled the initial SQL data bind feature.
- **Initial SQL query** - select the initial SQL data bind query. The query should return a single data row and each column within the data row should be called the short field name of the form field.
- **External DB Connection** – enter the code for establishing the connection to an external database

7.42 Setting up the Payment Gateway

The Payment Gateway feature is a few enhnacement added within the 2.6 release of Dynamic Forms. This feature will allow you to process payments using Dynamic Forms, offering the ability to use the module as a ecommerce solution and single-form payment solution. Within the current version the module supports the following payment methods:

- **Authorizenet Gateway** – see section 7.43
- **PayPal Gateway** - see section 7.44
- **PayFlow Pro** – see section 7.45
- **Verifi** – see section 7.46

The payment gateway feature (when enabled) will process payments after form validation but before any form completion events. This will allow you to validate the form fields before processing a payment, but only process any completion events such as adding the user to a role, sending out an email, or any other processing takes place.

To view a demonstration of Dynamic Forms being used as a payment gateway gateway you can visit this Dynamic Forms w/ Authorizenet Tutorial.

In order to start setting up the payment gateway, choose option “Payment Gateway” after clicking on “Module Configuration”.

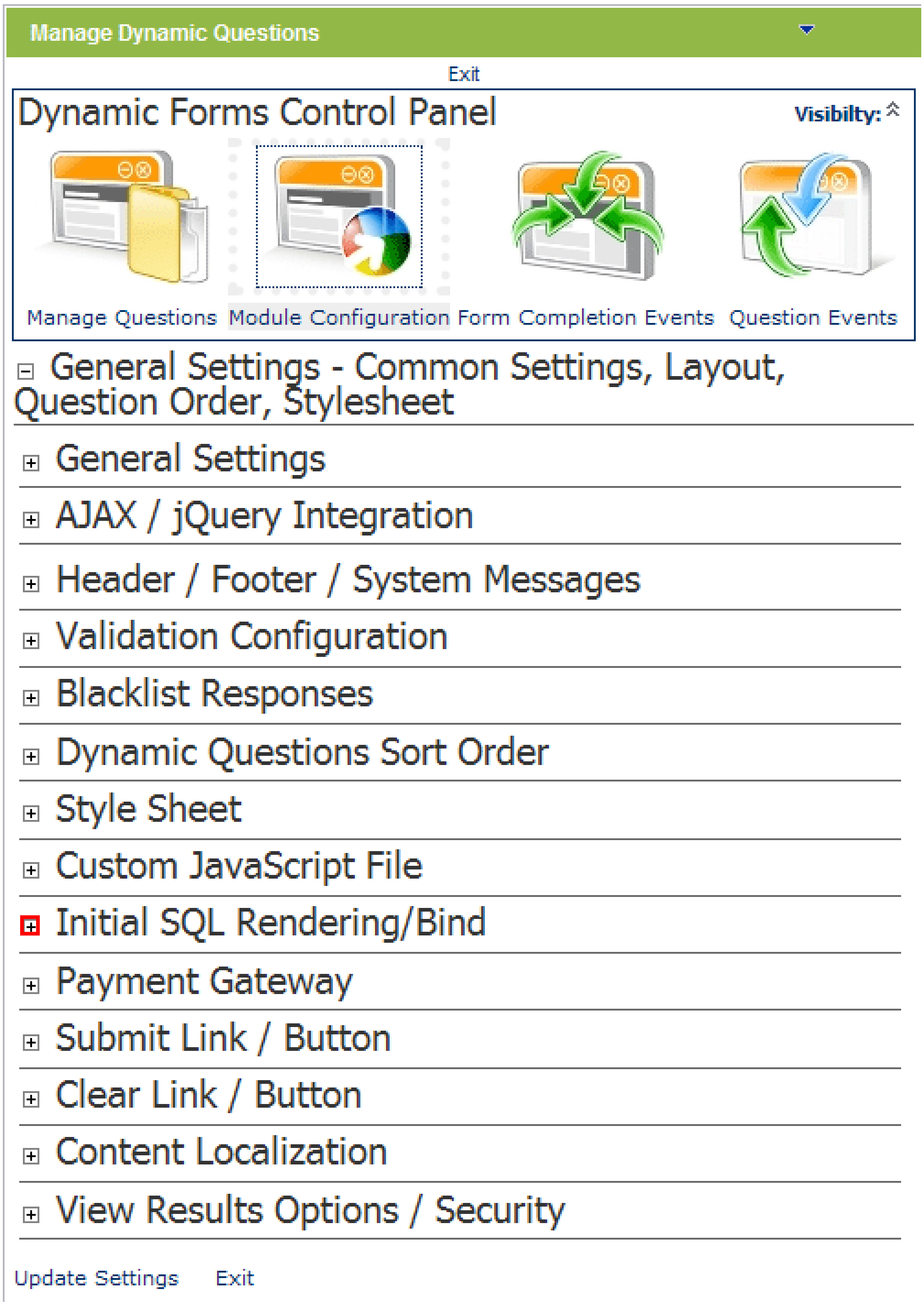


Figure 129: Setting up the payment gateway

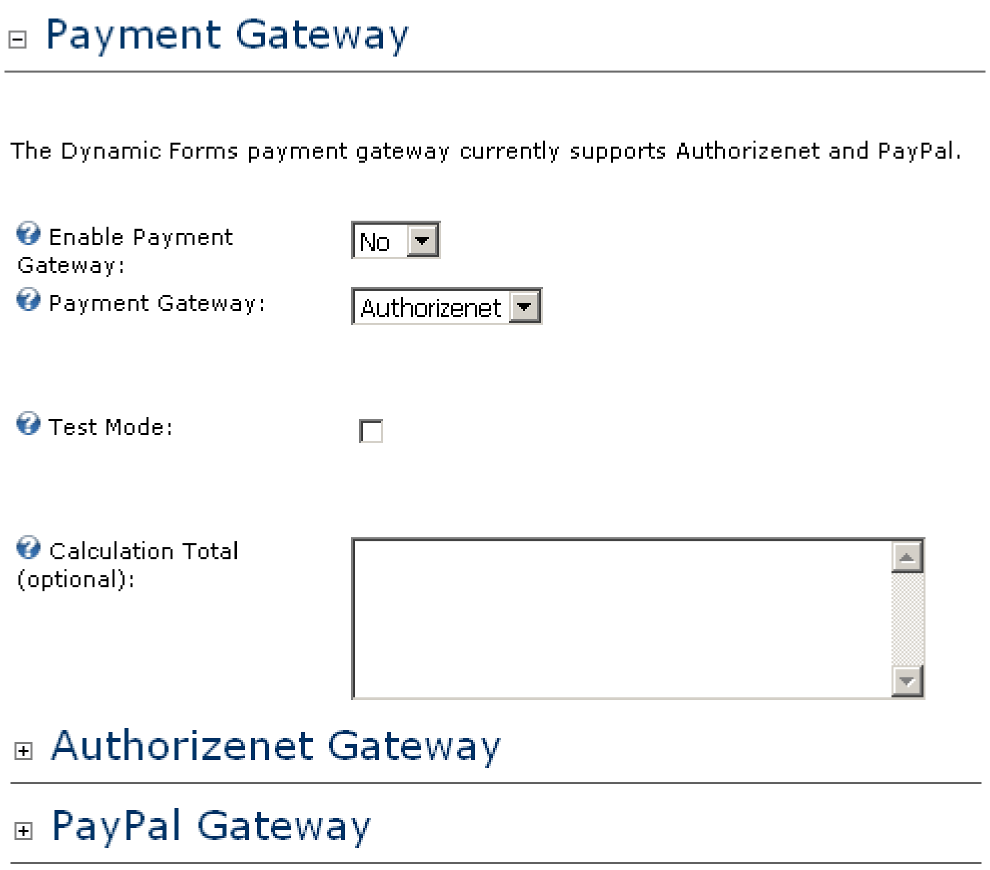


Figure 130: Setting up the payment gateway

The following parameters are available:

- **Enable Payment Gateway** – choose "Yes" to enable the payment gateway or "No" to disable it
    - **Note:** You can create a field with a short field name called 'EnablePGateway' to allow the user to choose for this feature to be enabled or not enabled. For this setting to operate correctly, you must still enable the gateway within this setting
    - The field option value should be 'True' to enable the gateway or 'False' to not enable the gateway.
- **Payment Gateway** – pull down menu for choosing the desired payment gateway
    - **Note:** You can create a field with a short field name called 'PGateway' to allow the user to choose if they want to pay via which Gateway.
        - For example, this could include a dropdownlist for Authorizenet (credit cards) or PayPal and allow the user to choose which payment gateway they would like to use.
        - The field option value should be 'Authorizenet' or 'PayPal' when being configured.
- **Test Mode** – select this checkbox to try the payment gateway functionality in test mode. Test mode is a feature allowed by Authorizenet which (even in production mode) will force the transaction to function within a testing environment and not actually process the transaction as a live transaction. This can be useful for testing as any responses you receive will be 'live', even if the transaction itself is now. For example, the system could return 'Invalid Credit Card Number' even in test mode.

- **Calculation Total (optional) –** This field can perform an optional calculation and then later be referenced as the parameter $(Calc_TotalField) within the Authorize net or PayPal additional parameters.

  Examples:

  - 10 * $(ProductPrice)
  - (5.00 * $(ProductQuantity)) + 10.00

  Note: You can add the querystring parameter &Debug=True to the URL (if friendly URLS /Debug/True/Default.aspx instead of /Default.aspx) in order to debug the field calculation that is being executed. You must be a portal administrator to use this feature.

  Note: New features have been added to 2.5 to also calculate checkboxlists and listbox multi selection fields. To use these you need to use parameters $(CalculateList:ShortFieldName). To set values within the checkbox group/listbox fields you need to set each list items value to be VALUE01_Price. You can have up to 20 items with values in it. For example VALUE01_19.99, VALUE02_9.99, VALUE03_29.99, VALUE04_19.99, VALUE05_9.99). By using this feature you can create total fields such as 'Check all items you want to purchase' or 'Check all add-on's you want to add' and Dynamic Forms will total all of the checked or selected items for you with this token.

- **Authorizenet Gateway** – for further information on how to setup the authorizenet gateway, see section 7.43
- **PayPal Gateway** - for further information on how to setup the authorizenet gateway, see section 7.44

7.43 Setting up Authorizenet Gateway

In order to start setting up the “Authorizenet Gateway”, choose option “Module Configuration” from the “Settings” page, click on “Payment Gateway” and then on “Authorizenet Gateway’.

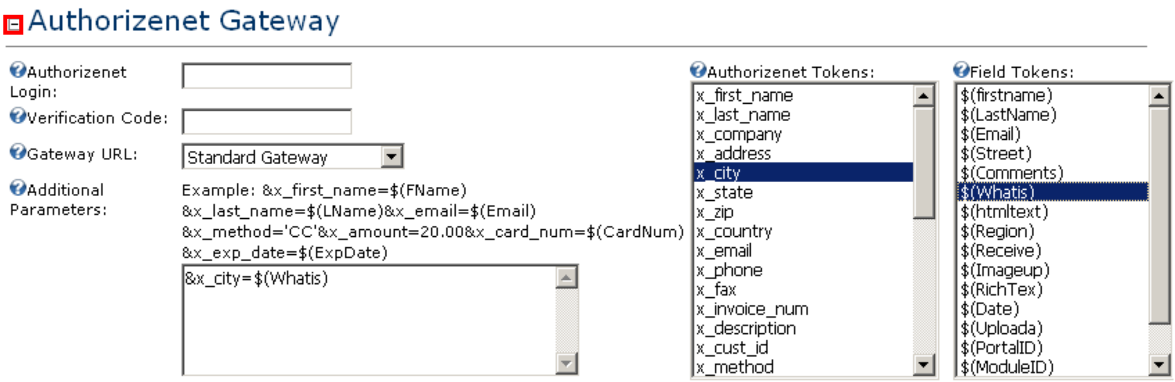


Figure 131: Setting up Authorizenet gateway

The following parameters are available:

- **Authorizenet Login** – use this field to enter your authorizenet username
- **Verification Code (Transaction Key)** – enter your Authorizenet transaction key. You receive this within Authorizenet by logging into the account settings and generating a transaction key (also known as API Transaction key or verification key)

- **Gateway URL** – Select the authorizenet gateway to use. The standard gateway (and most commonly used) is https://secure.authorize.net/gateway/transact.dll and the Developer Test Gateway is https://test.authorize.net/gateway/transact.dll. The developer test gateway should only be used if you are a developer and do not actually have a live Authorizenet account.

- **Additional parameters -** These additional parameters are passed. Within the Authorizenet section you will notice two listbox's. One that includes Authorizenet Tokens and another that includes Dynamic Tokens that are available from Dynamic Forms fields. The additional parameters can pass as many (or as few) parameters to Authorizenet for processing. This 'advanced method' allows the most flexibility for your implementation.

  Note: There are a few parameters that MUST be included or else Authorizenet will simply not work. At a minimum the credit card number, expiration date, method type (i.e. credit card), and amount MUST be set. Items such as currency and all of the billing contact info are optional. Setting these parameters follows a very generic and standard method. These should be &AuthorizenetToken=SomeValue

  Note: Selecting a token from the Authorizenet token listbox or Dynamic Fields listbox will automatically move that token over to the additional parameters textbox in the standard format of &AuthorizenetToken=

Examples:

- To specify the first name, last name, email, credit card number, credit card expire date, and the amount, the additional parameters would be:

  &x_first_name=$(FName)&x_last_name=$(LName)&x_email=$(Email)&x_method='CC'&x_amount=20.00&x_card_num=$(CardNum)&x_exp_date=$(ExpDate)

- This example just displays using the calculation field for the amount:

  &x_first_name=$(FName)&x_last_name=$(LName)&x_email=$(Email)&x_method='CC'&x_amount=$(Calc_TotalField)&x_card_num=$(CardNum)&x_exp_date=$(ExpDate)

- This example makes the billing reoccurring and changes the system from credit cards to checks, also collects the users zip code, passes the IP Address along to Authorizenet, and also collets the users drivers license number.

  &x_first_name=$(FName)&x_last_name=$(LName)&x_email=$(Email)&x_method='ECHECK'&x_amount=$(Calc_TotalField)&x_bank_name=$(BankName)&x_bank_acct_type=$(AcctType)&x_bank_aba_code=$(BankABACode)&x_bank_acct_num=$(BankAccountNumber)&x_recurring_billing="TRUE" &x_drivers_license_num=$(AnotherField)&x_ip_address=$(IPAddress)

- **Authorizenet Tokens** – This is the full list of available Authorizenet tokens that can be passed information for this transaction

- **Field Tokens** – This is the full list of available Dynamic Tokens that can be parsed. This list is based on the short field names of dynamic fields and other internal tokens such as portal ID, UserID, IP Address, and others.

Tips:

- Don't forget that Authorizenet requires SSL to be setup on your site for the system to operate in a production or live environment.
- If you are curious what is happening behind the scenes, you can add the &Debug=True to your URL when processing payments to determine the exact post that module is sending to Authorizenet. You must be a site administrator or host user to review this.

- To can review a live demonstration of Dynamic Forms with Authorizenet from the following tutorial/demonstration: [Dynamic Forms w/ Authorizenet Tutorial]

7.44 Setting up PayPal Gateway

In order to start setting up the "PayPal Gateway", choose option "Module Configuration" from the "Settings" page, click on "Payment Gateway" and then on "PayPal Gateway". The following screen will be displayed.

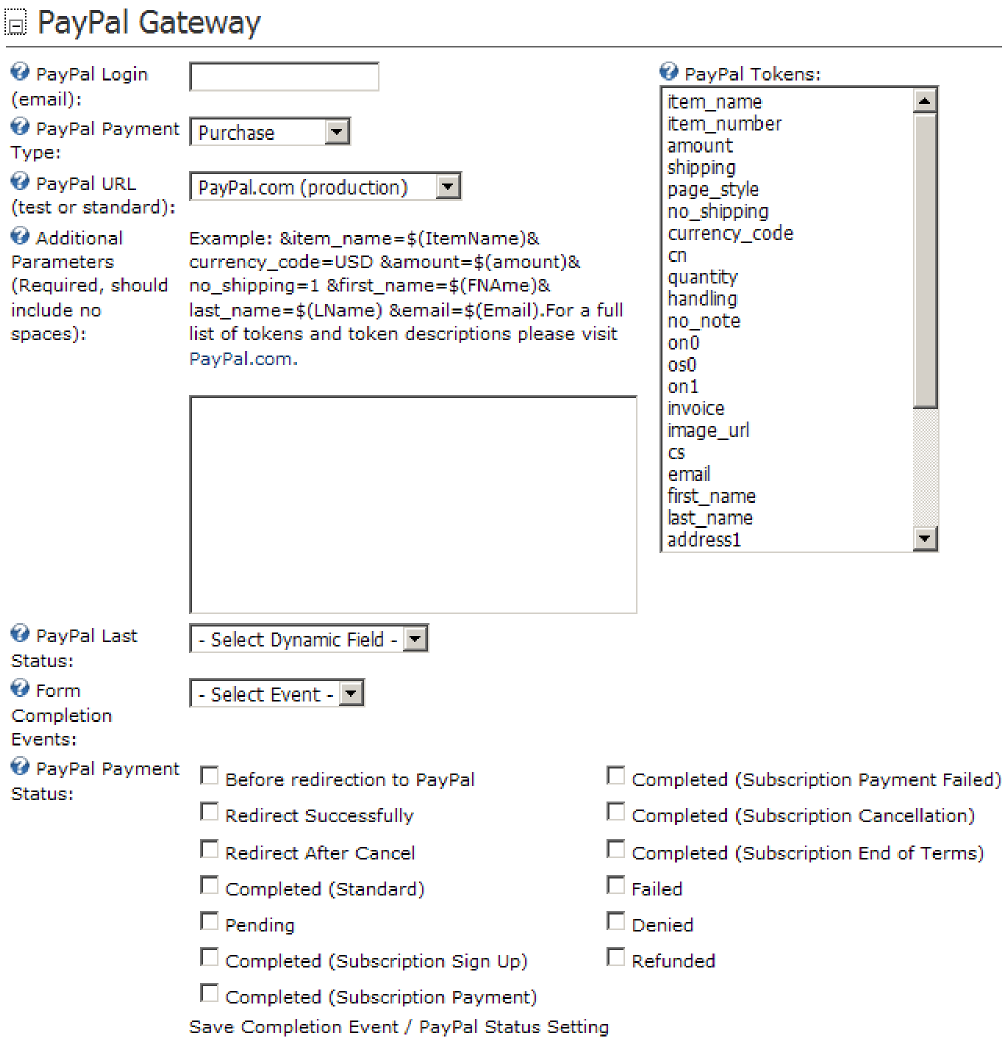


Figure 132: Setting up PayPal Gateway

The following parameters and options are available inside this screen:

- **PayPal Login (email)** – enter the email address connected to your PayPal account
- **PayPal Payment Type** – choose the payment type between the following:
    - **Purchase** – the 'Purchase' payment type represents single instance purchases within this form/payment; you will want to add additional parameters for the amount of the purchase and the contact info for the user. For a full list of variables that affect **[Purchases or 'Buy Now' payments click here.]**
        - **Note:** This payment type represents the x_click PayPal payment cmd variable
    - **Subscription** – the subscription payment type represents a recurring purchases subscriptions. You will want to add additional parameters for the subscription rate, billing cycle, and billing unit. These parameters represent A3, P3, and T3 PayPal

variables. For example: $10.00 (rate), every 3 (units) months (billing cycle). For a full list of available variables that affect **subscription payments click here.**

- **Note:** This payment type represents _xclick-subscriptions PayPal payment cmd variable

- **PayPal URL (test or standard)** – choose if you would like the payment to be processed by the standard PayPal gateway or the sandbox gateway
  - **PayPal.com (Production)** – choose this to use PayPal live gateway
  - **Sanbox.PayPal (test)** – choose this to use PayPal test gateway
    - **Note:** The PayPal Sandbox gateway requires you to setup a separate PayPal account at http://sandbox.paypal.com.
    - During testing and setup of the PayPal Payment Gateway within Dynamic Forms, it is highly suggested to setup a SandBox environment for testing of your form payments.
- **Additional Parameters (Required, should include no spaces)** – use this field to enter the additional parameters for this PayPal transaction; this includes passing dynamic tokens for each optional PayPal token such as the users first and last name, the amount, the item name, etc.
  - PayPal tokens represent variables that you can pass to PayPal for the transaction. Within the additional parameters these tokens should be references as &PayPalVariable=
  - Dynamic tokens represent the short field names for the Dynamic Fields that you have setup within Dynamic Forms. These fields should be referenced as $(ShortFieldName)
  - Example: Let's assume that you wanted to accept payments and you wanted the user to enter the amount that they wanted to pay in a field you setup with a short field name of 'MyAmount'. Additionally, standard contact fields are setup such as name and address, set the currency to be USD, and disable shipping within this purchase.
    - The additional parameters would look something like this:
    - &amount=$(MyAmount)&currency_code=USD&item_name=YourProductName&no_shipping=1&first_name=$(FName)&last_name=$(LName) &email=$(Email)
  - Example 2: Let's assume that you wanted to accept a subscription recurring payment for 10 dollars every 1 month. Additionally, standard contact fields are setup such as name and address.
    - The additional parameters would look something like this:
    - &a3=10.00&p3=1&t3=M&no_note=1&currency_code=USD&item_name=YourProductName&no_shipping=1&first_name=$(FNAme)&last_name=$(LName) &email=$(Email)
- **PayPal Tokens:**
- The following variables can also be found at: https://www.paypal.com/IntegrationCenter/ic_std-variable-reference.html
- The following tokens are required as additional parameters for single purchases:
  - Amount
  - Item_name
- The following tokens are required as additional parameters for subscription purchases:
  - **A3** (represents subscription rate)
  - **P3** (represents billing cycle.) Such as '3' for every 3rd instance of the setting T3.

- **T3** – (represents billing cycle units). This is the units of the regular billing cycle (p3, above) Acceptable values are: D (days), W (weeks), M (months), Y (years).
- **No_note** – This must be set to 1 as PayPal does not support allowing a note to a subscription

- **PayPal Last Status** - select this field if you would like to represent the status field from the last PayPal IPN or status
- **Form Completion Events** - this dropdownlist represents the form completion events that have been setup for this form; within this configuration area you can determine which completion events will be executed based on which specific payment status as returned from PayPal. When the PayPal payment gateway is enabled, no completion events will be fired off until payment has been completed and are linked to these payment status codes.
  - **Note:** keep in mind that often these notifications from PayPal can happen multiple times throughout the users transaction and do not necessarily reflect when the user is returned to the form after payment; for example, if a user cancels their subscription months from the initial signup, the PayPal system will send notification back to your Dynamic Forms module and it will then execute the appropriate completion events to fire off.

- **PayPal Payment Status** - check the status which will enable this completion event to be executed
  - Redirect Successfully
  - Redirect After Cancel
  - Completed (Standard)
  - Pending - Often if the user uses an ECheck option
  - Completed (Subscription Sign Up)
  - Completed (Subscription Payment)
  - Completed (Subscription Payment Failed)
  - Completed (Subscription End of Terms)
  - Completed (Subscription Cancellation)
  - Failed
  - Denied
  - Refunded
- To can review a live demonstration of Dynamic Forms with PayPal® from the following tutorial/demonstrations:
  - Dynamic Forms PayPal® Integration for Purchase Demonstration
  - Dynamic Forms PayPal® Integration for Subscription Demonstration
  - Dynamic Forms PayPal Integration (optionally turn payment gateway off based on short field name ‘EnablePGateway’)
  - Dynamic Forms PayPal® Integration (optionally switch between PayPal and Authorizenet payment gateways based on short field name of ‘PaymentGateway’)

7.45 Setting up PayFlow pro

In order to start setting up the “PayFlow Pro”, choose “Module Configuration” from the “Settings” page. Then click on “Payment Gateway” and choose “PayFlow pro”. The following screen will be displayed.

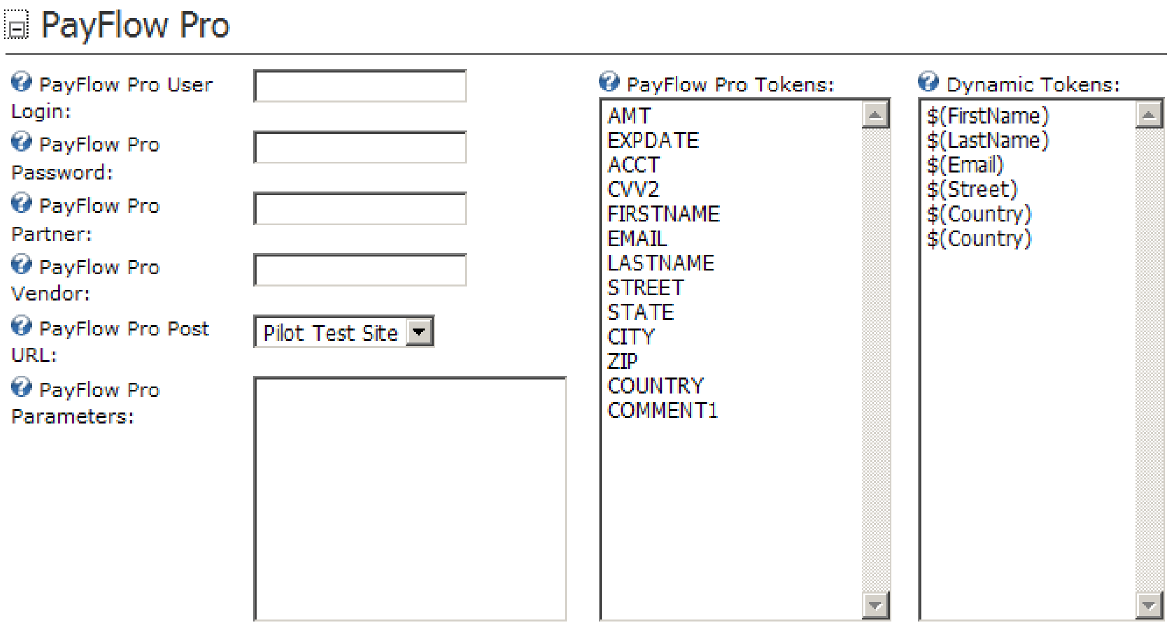


Figure 133: Setting up PayFlow pro

The following parameters and options are available inside this screen:

- **PayFlow Pro User Login** – enter the email address connected to your PayFlow Pro
- **PayFlow Pro Password** - enter the password connected to your PayFlow Pro
- **PayFlow Pro Partner** - enter your partner information for Pay Flow Pro account
- **PayFlow Pro Vendor** - enter your vendor information for Pay Flow Pro account
- **PayFlow Pro Post URL** - specify the parameters that are passed to this payment gateway
- **PayFlow Pro Parameters** – specify the PayFlow Pro parameters

7.46 Setting up Verifi

In order to start setting up the "Verifi", choose option "Module Configuration" from the "Settings" page. Then, click on "Payment Gateway" and choose "Verifi". The following screen will be displayed.

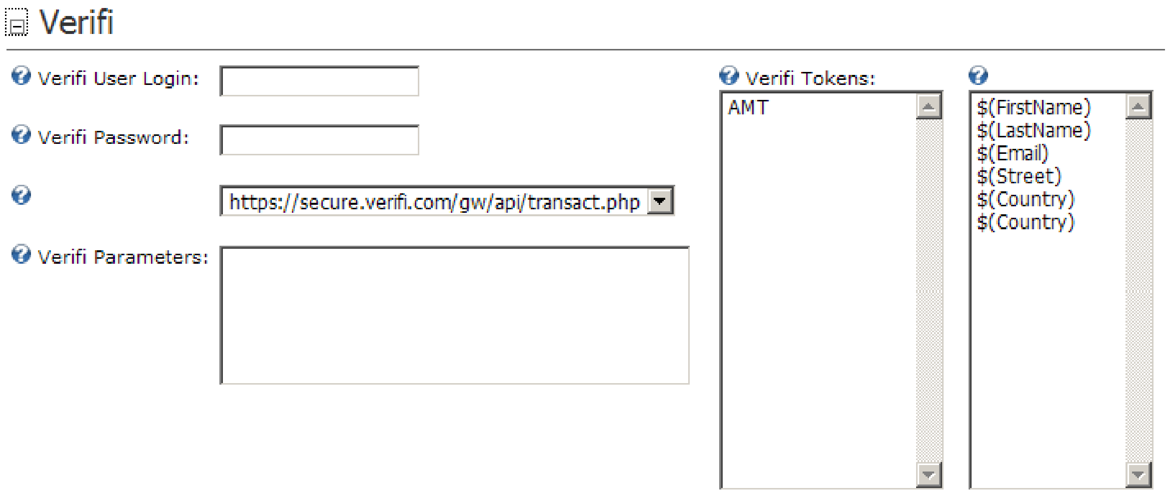


Figure 134: Setting up Verifi Gateway

The following parameters and options are available inside this screen:

- **Verifi User Login** – enter your Verifi username
- **Verifi Password** - enter your Verifi password
- **Verifi Parameters** – specify Verifi parameters
- **Verifi Tokens** – select the desired Verifi tokens

7.47 Setting the Submit Link or Button

This option is used for choosing whether the “submit” option for your form will be text or image and if it is image, which one should be used. In order to start setting the link or button, choose option “Submit link/Button” inside the “General Settings” page.

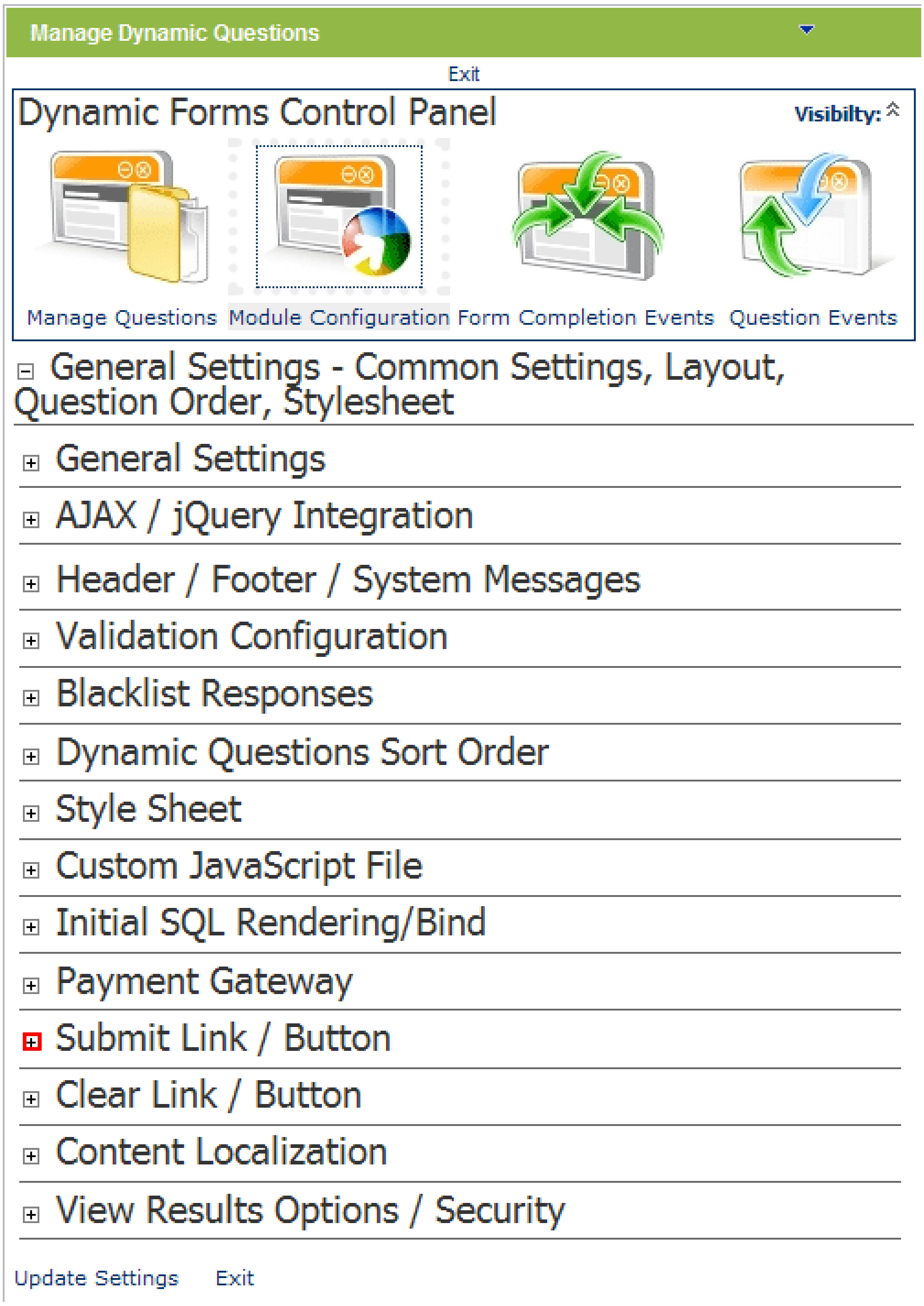


Figure 135: Setting submit link or button (step 1/2)

The following screen will be displayed.

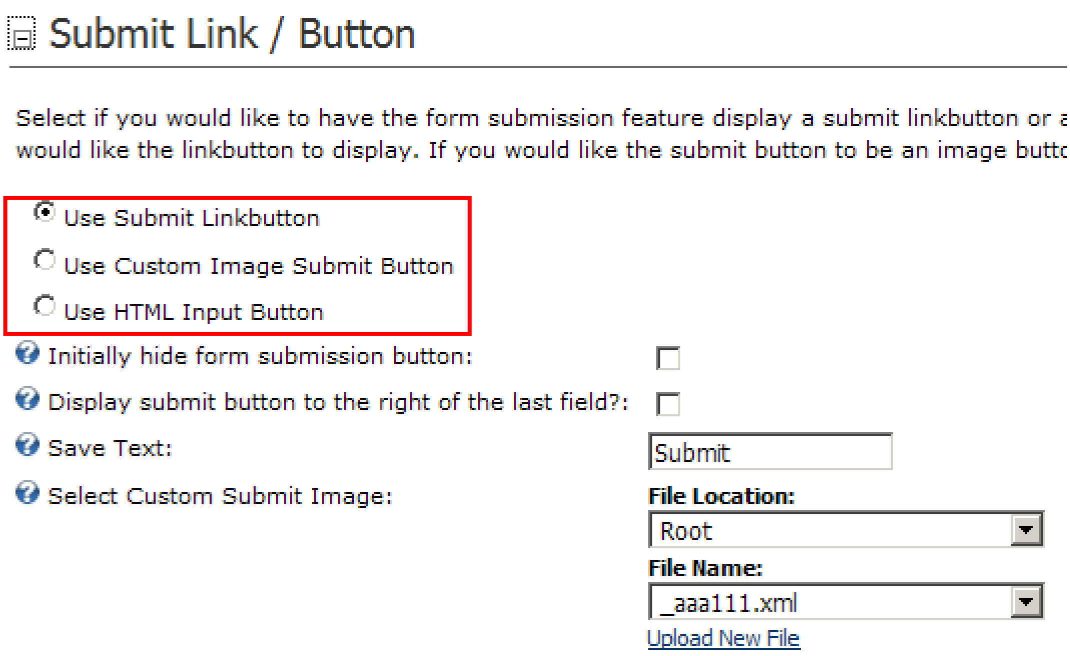


Figure 136: Setting submit link or button (step 2/2)

The following parameters are available inside this screen:

- **Use Submit link Button** – if you choose this option the text link will be displayed below the form
- **Use Custom Image Submit Button** – select this radio button in case you wish to use a custom image for the submit option
- **Use HTML Input Button** - select this radio button in case you wish to use a standard HTML Input Button for the submit option
- **Initially hide form submission button** – by enabling this feature the submission link/image button will initially be hidden. This feature works with the Client Side Events feature to then display the hidden submission buttons.(For example, hide the submit button until the user clicks the terms and conditions) View Demonstration
- **Display submit button to the right of the last field** - by selecting this feature the submit link, image, or html input button will be displayed to the right of the last field (same table row) instead of at the very bottom of the form
- **Save Text** – enter the desired text for the submit link in case you have chosen that option (**e.g.** “Submit”)
- **Select Custom Submit Image** – choose the desired image file from this pull down menu after choosing option “Use Custom Image Submit Button”

After setting the desired parameters, click on the "Update Settings” button to save the changes.

7.48 Clear Link/Button

This option is used for including a reset i.e. clear button within your form, which the users can use in order to start filling it from the beginning.

The clear button will appear directly next to the form submission button and will reset the form to its initial results when the form was loaded. The form clear or reset button can optionally be a link button, or an image button, and you can specify the clear text for the link button or image below.

The reset button can either be setup to use AJAX / Javascript client side code or can be simply refresh the page. The AJAX method is faster but is not available for forms that utilize features such as server side dynamic question events.

In order to start defining this button, click on the "+" symbol next to the "Clear Link / Button" label.

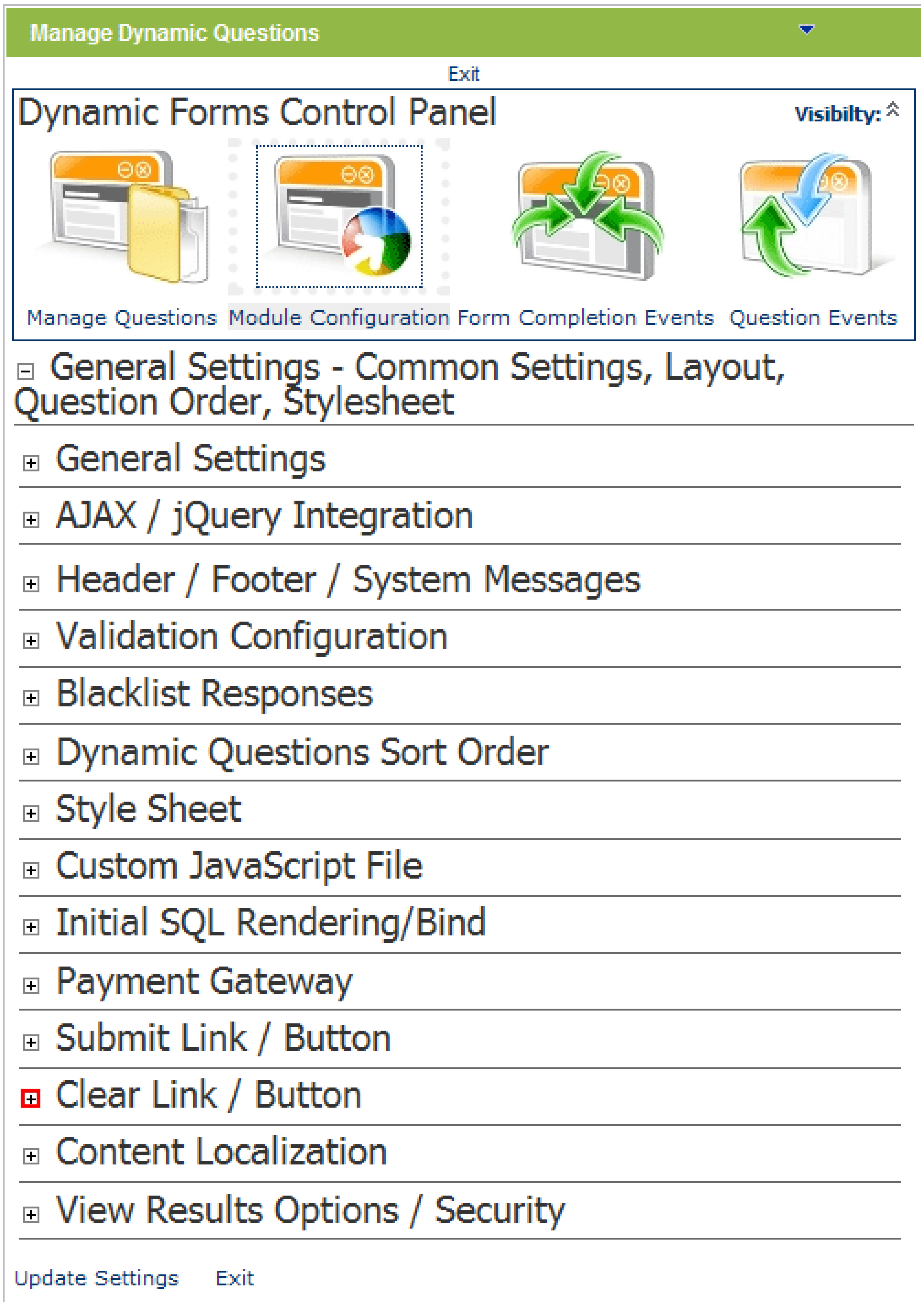


Figure 137: Defining the Clear Link / Button (step 1/2)

The following screen will be displayed.

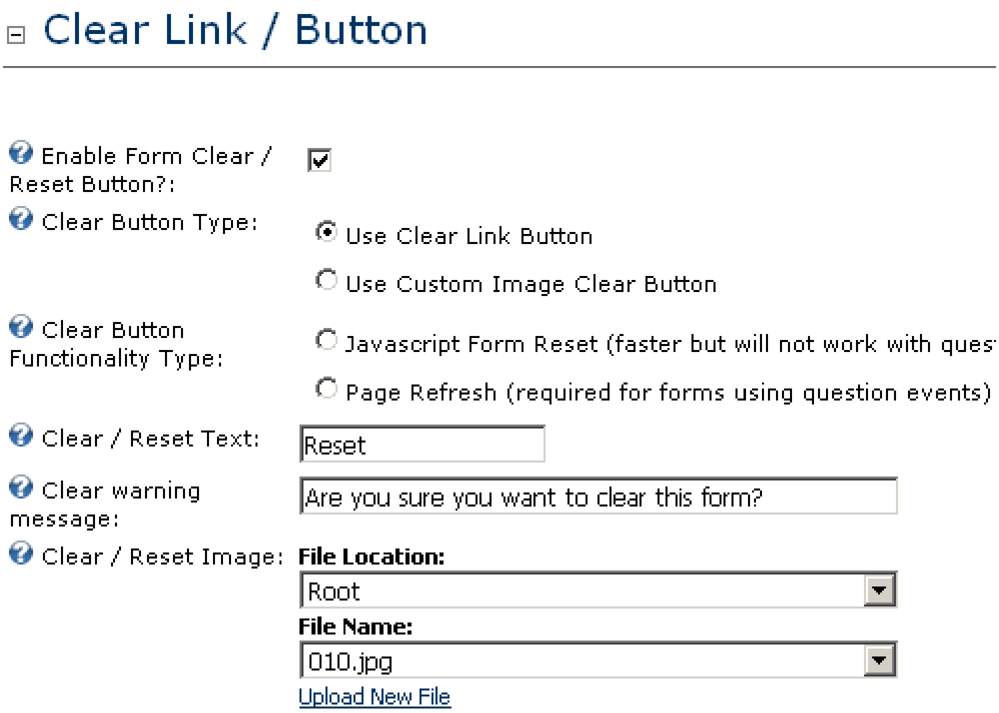


Figure 138: Defining the Clear Link / Button (step 2/2)

The following parameters and options are available inside this screen:

- **Enable Form Clear / Reset Button** - select if you would like to enable a clear link or image button for your form
- **Clear Button Type** – select the desired type for the “Clear” button
    - **Use Clear Link Button** – select this option if you only want to use a link
    - **Use Custom Image Clear Button** – select this option if you wish to define a custom image for the clear button (**note**: you can set the desired image under “Clear/Reset Image”)
- **Clear Button Functionality Type -** select the clear button functionality type for this form
    - **The javascript type** is faster but will only work on forms that do not use postback / question events.
    - **Page Refresh** (required for forms using question events) - the redirect or page refresh page will not be as fast but is required for forms that use question events
- **Clear / Reset Text** – enter the text for the link in case you wish to display the link (**e.g.** “Reset”)
- **Clear warning message -** the clear warning message is an optional message pop up that can ask the user if they are sure they want to reset the form
    - **Note:** the clear warning message will not fire if this setting is left blank or if the page refresh option is checked (only the javascript reset feature includes this optional pop up message)
- **Clear / Reset Image** – use these pull down menus to select the image in case you wish to use a custom image for the reset function
    - **Note:** choose “Upload New File” to upload an image from your PC

The screenshots below demonstrate each of the 3 cases.

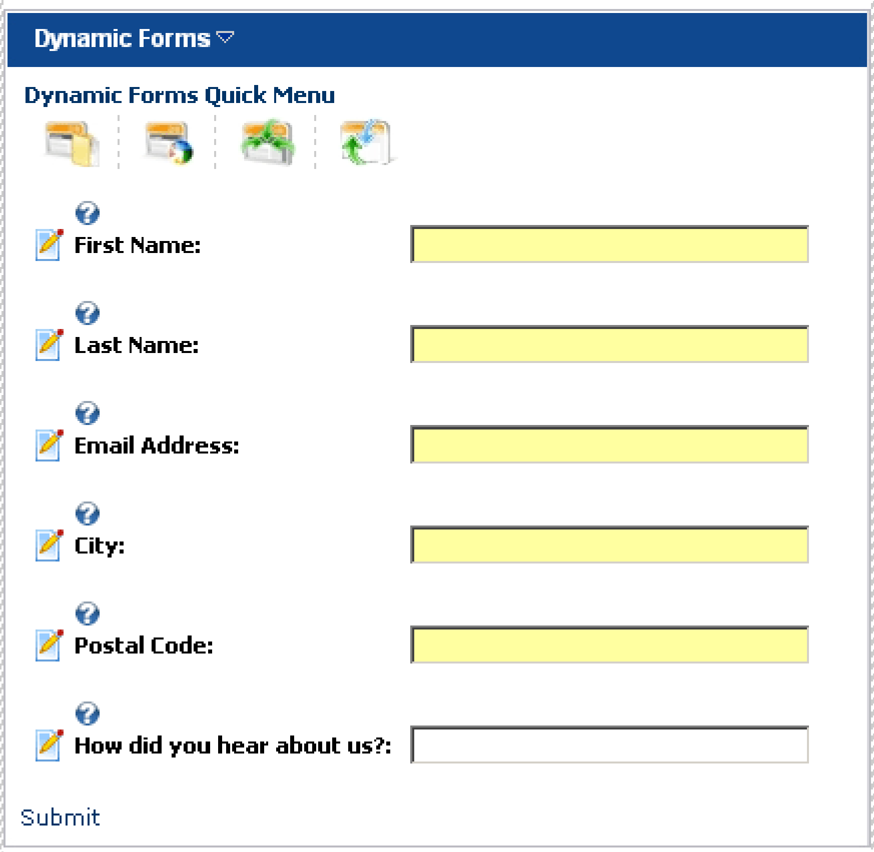


Figure 139: The form when the reset function has not been enabled

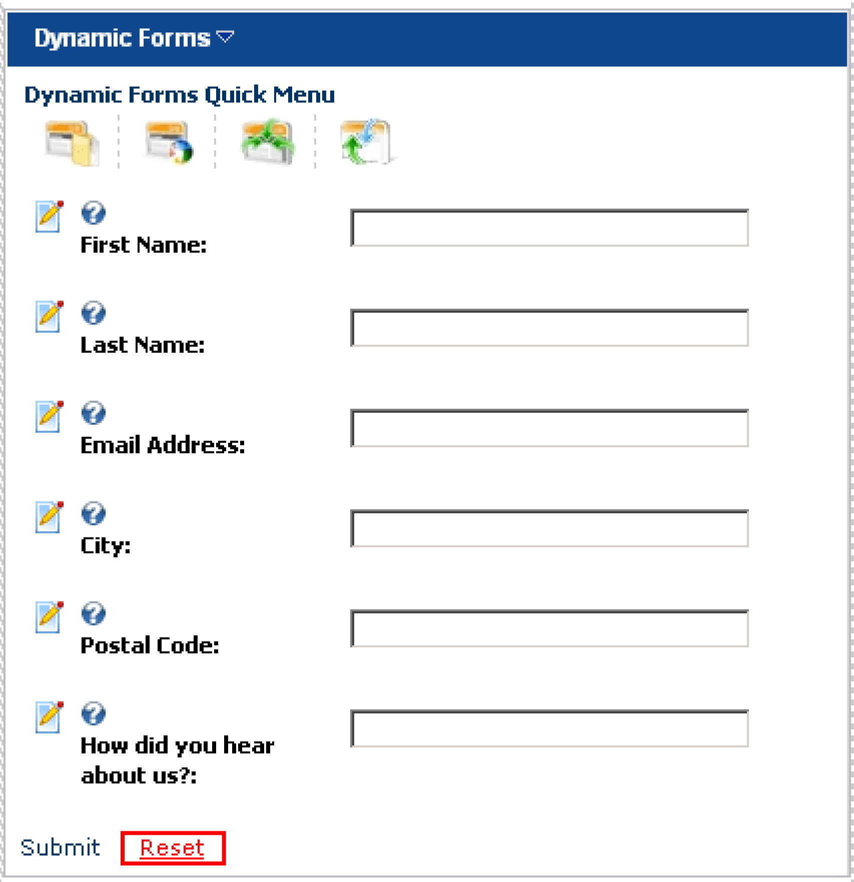


Figure 140: The form when the reset function has been setup as a textual link

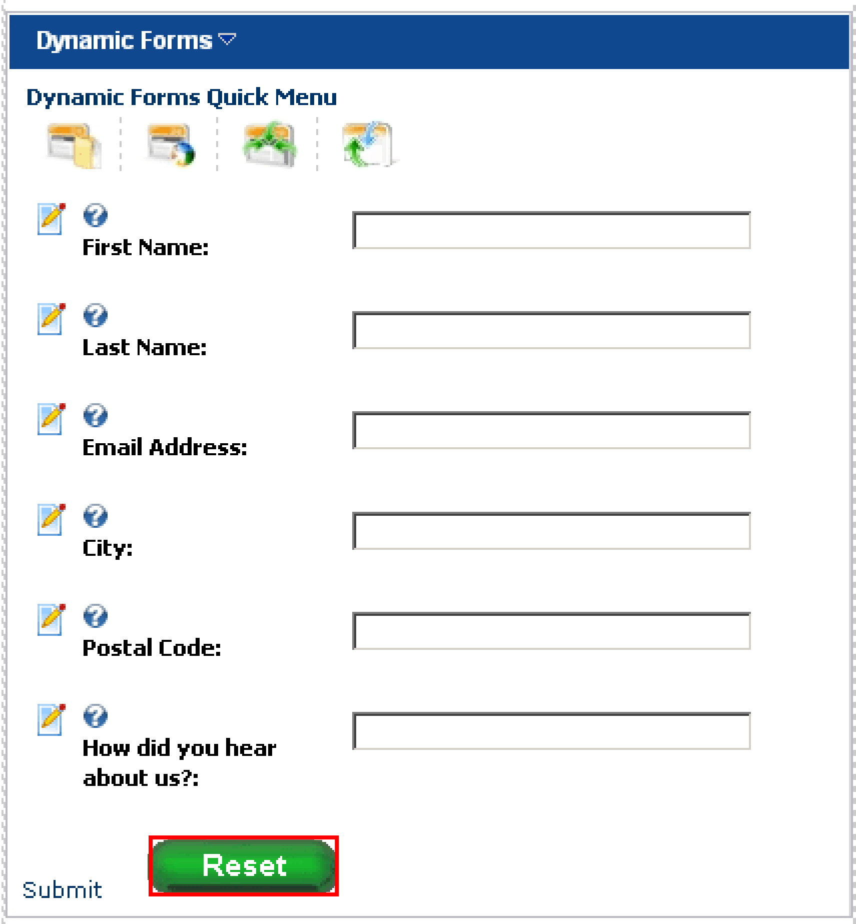


Figure 141: The form when the reset function has been setup as a button

7.49 Managing Content Localization

The content localization feature allows you to setup different form headers/footers/submit button text or button depending on the language the user has tied to their account or chosen on the site form the list of available languages

In order to start managing the content localization, click on the "+" symbol next to the "Content Localization" label after choosing the "Module Configuration" option.

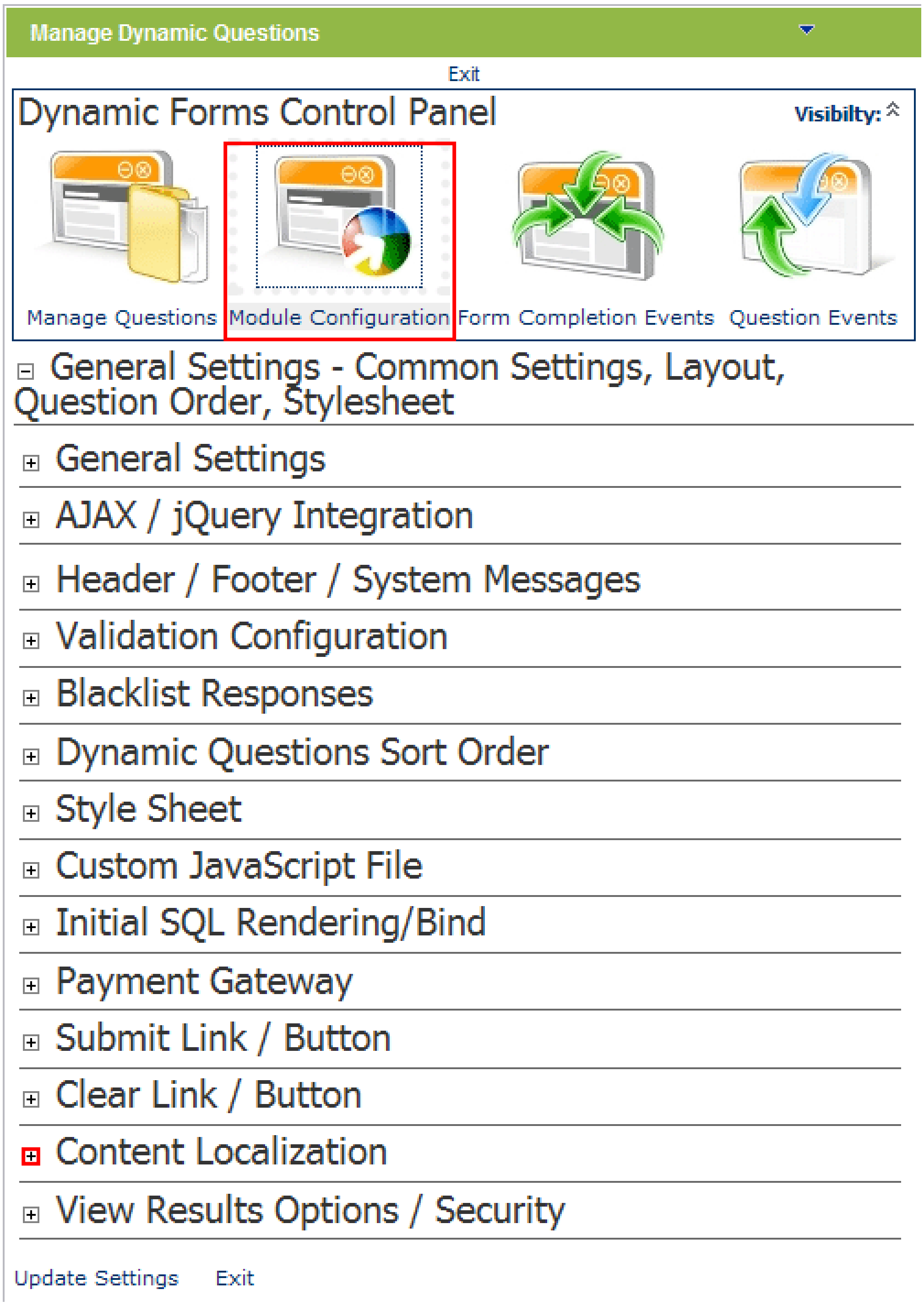


Figure 142: Managing Content Localization

The following options are available inside this screen:

- **Form Localization** – option for localizing the form i.e. displaying the elements of the form in different languages based on the end user choice of language (see section 7.49.1)

- **Question Localization** – option for localizing the questions i.e. displaying the questions in different languages based on the end user choice of language (see section 7.49.2)

7.49.1 Managing Form Localization

The purpose of the "Form Localization" option is to allow you to define form elements in different languages based on the end user choice of language.

In order to start localizing the form, click on the "+" symbol next to the "Form Localization" label. **Note:** To enable languages for your portal please refer to the DotNetNuke® User Guide. You can install language packs under Admin, Languages. Once you have enabled additional language packs within your portal those languages will be available from the dropdownlist below.

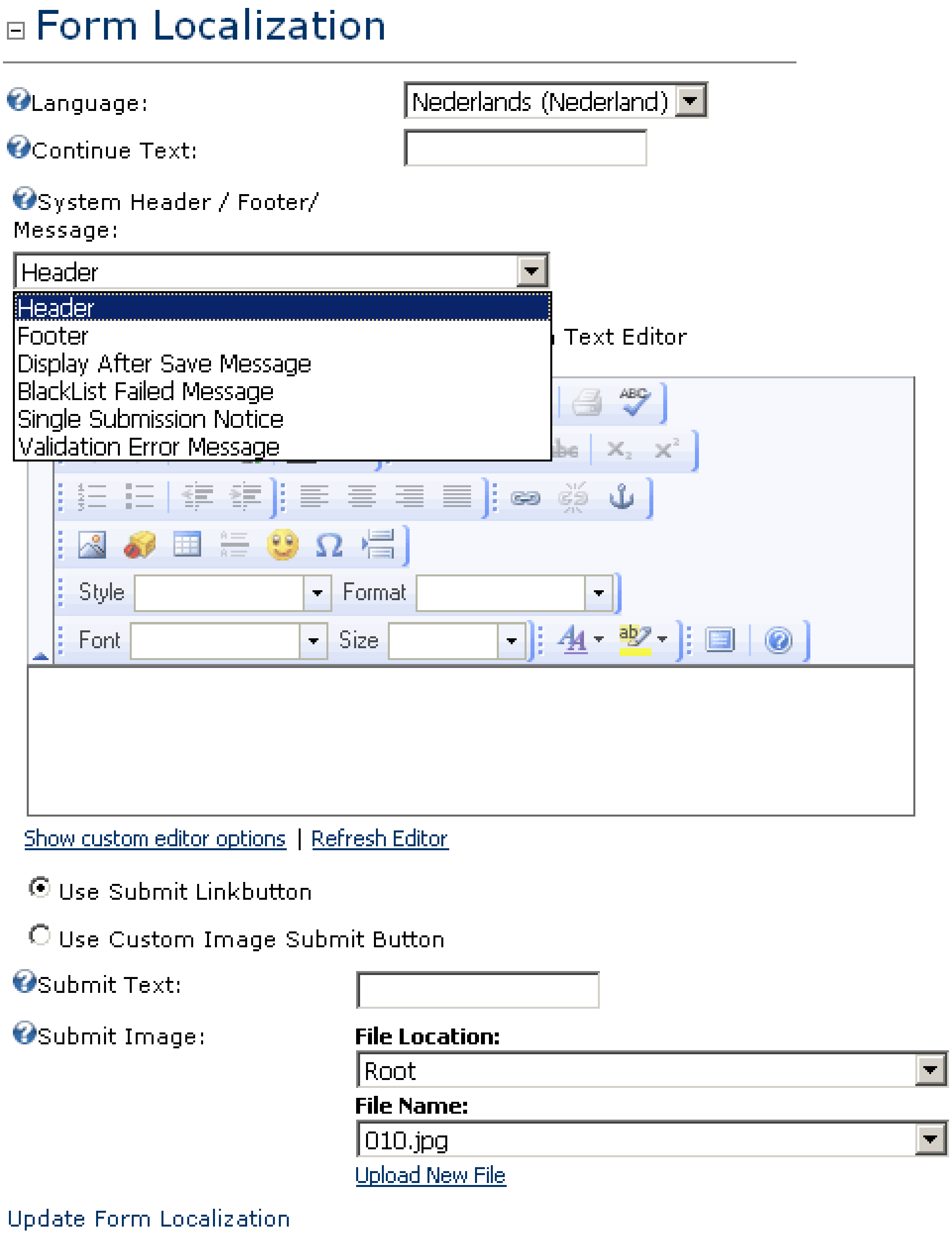


Figure 143: Managing form localization

The following parameters are available inside this screen:

- **Language** – choose the language this form will be related to; i.e. once the user chooses the desired language, the content localization connected to that language will be displayed
- **Continue Text** – enter the text that will serve as a "Continue" text

- **System Header / Footer/ Message** – select the desired option from the pull down menu to define header, footer or message displayed after the form has been submitted in the desired language
- **Use Submit Linkbutton** – select this option if you want to use the standard submit button or link
- **Use Custom Image Submit Button** – select this option in case you wish to upload a custom submit image
- **Submit Text** – enter the text which will be displayed within the standard submit button (**note:** in case you have chosen “use submit linkbutton” option)
- **Submit Image** – select the custom image by using the “File Location” and “File Name” pull down menus or upload the new file by clicking on the “Upload New File” link

After setting the desired parameters, click on the "Update Form Localization” link.

7.49.2 Managing Question Localization

The purpose of the “Question Localization” option is to allow you to define questions in different languages based on the end user choice of language.

In order to start localizing the questions, click on the “+” symbol next to the “Question Localization” label. The following screen will be displayed.

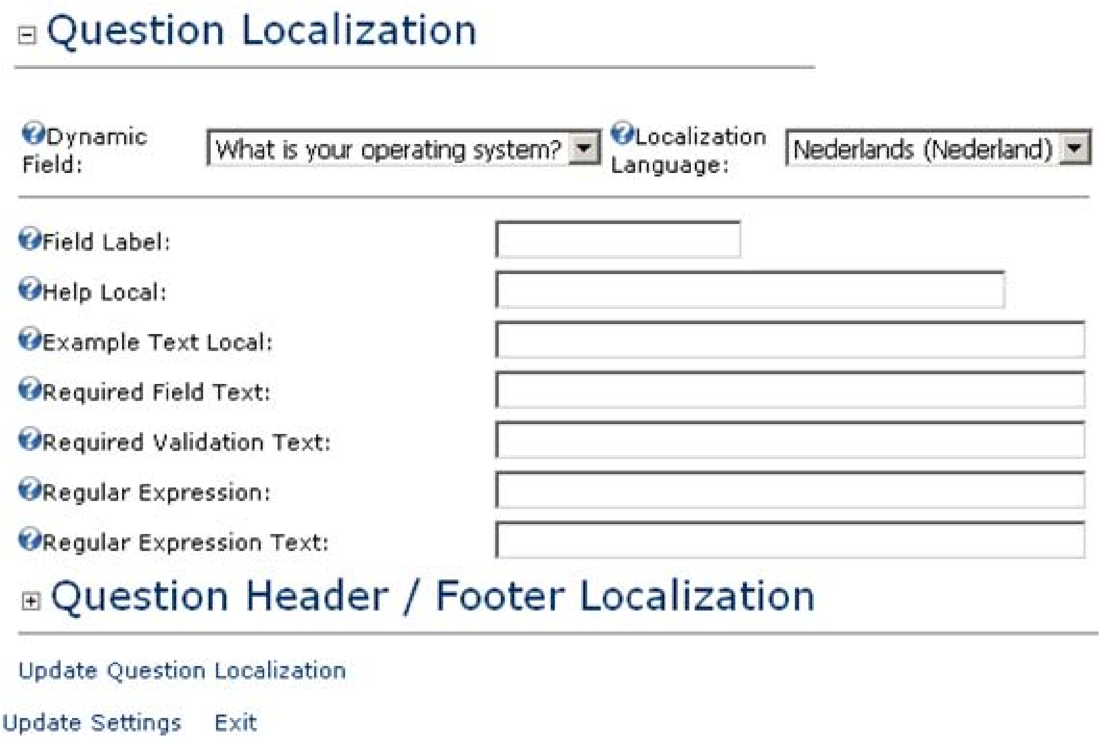


Figure 144: Managing Question Localization

The following parameters are available inside this screen:

- **Dynamic Field** – choose the dynamic question you wish to define a different language for
- **Localization Language** – choose the language this question will be connected to
- **Field Label –** enter the question label in the desired language
- **Help Local** – enter the help text in the desired language
- **Example Text Local** – enter the example text in the desired language
- **Required Field Text** – enter “Required field” text in the desired language
- **Required Validation Text** – enter the required validation text in the desired language

- **Regular Expression** – enter the help text in the desired language
- **Regular Expression Text** – enter the regular expression text in the desired language
- **Question Header/Footer Localization** – click on the "+" symbol next to this label to define question header and/or footer
- ***Tip:*** If you want to localize options for combo box, radio buttons, checkbox groups, or listbox's you will need the options to be included within a table and use the 'SQL Options' option when setting up the options. You would need to create a column within the table to reference the values for a language. Here is an example:

  Select Text as QuestionOption, Text as QuestionOptionValue from YourTable where Language = '$(CurrentLanguage)'

  The parameter to pull the users current language is $(CurrentLanguage). This example above demonstrates pulling options from a table called 'YourTable' with two columns; One column called 'Text' and another column called 'Language'.

  For more information on this topic please refer to the Data Springs Product Forums.

After setting the desired parameters, click on the "Update Question Localization" link to save the changes. You can repeat this procedure for all questions inside the form. The screenshot below demonstrate the procedure of choosing a different language.

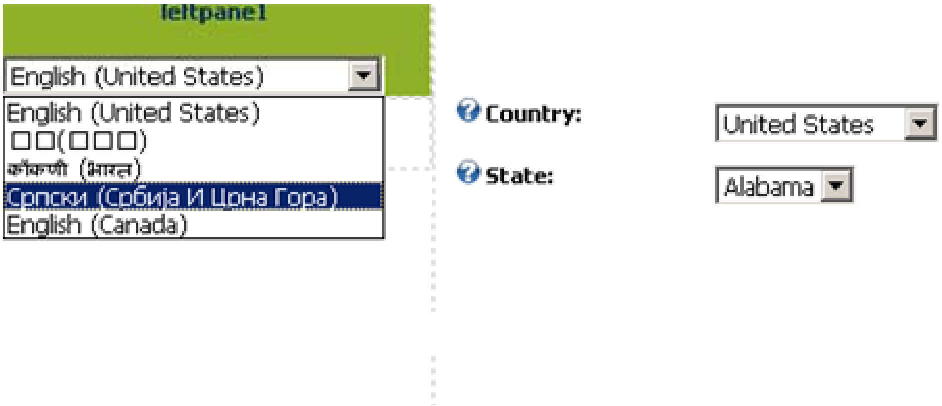


Figure 145: Choosing a desired language

The screenshot below demonstrates the form with the localized questions.

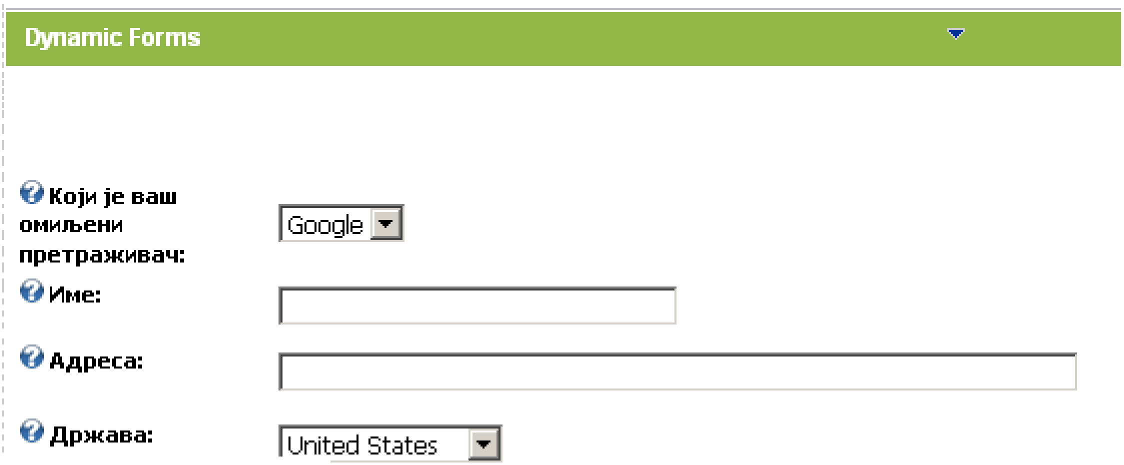


Figure 146: Localized questions

7.50 Managing the access rights for form results

The users with the “view” (non-admin role) rights can also be allowed to view, edit, manage templates, export, and purge form results.

In order to start managing the rights for accessing these options, click “Module Configuration” and then choose “View Results Options/Security”.

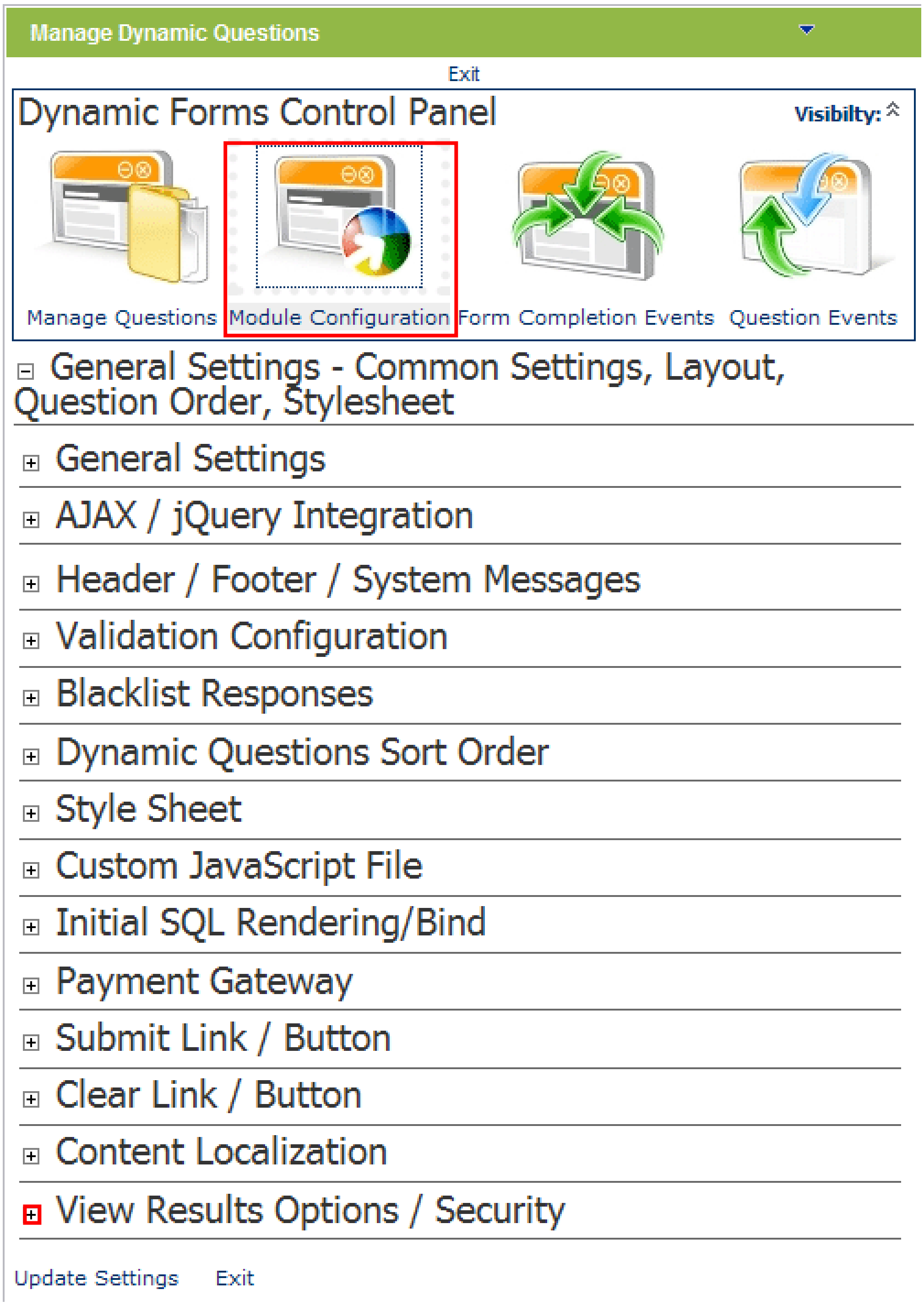


Figure 147: Managing the View Results Options and Security (step 1/2)

The following page will be displayed.

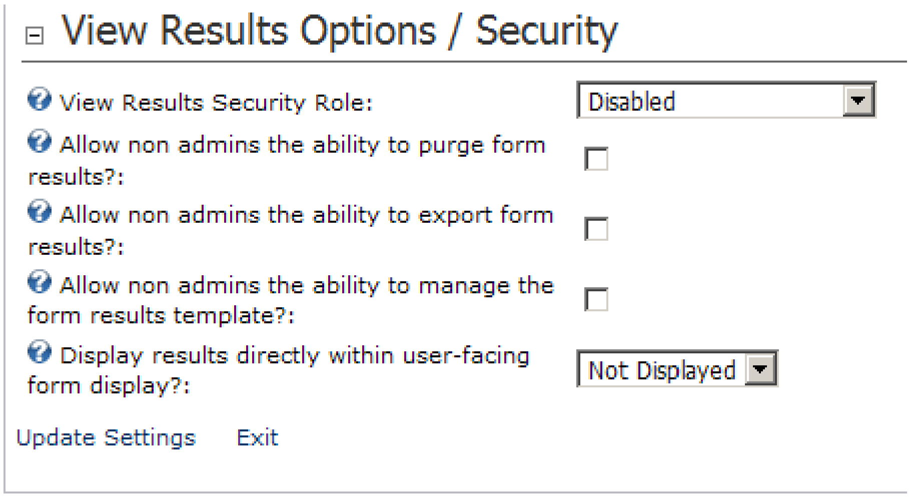


Figure 148: Managing the View Results Options and Security (step 2/2)

The following options and parameters are available:

- **View Results Security Role** – use this option to select the security role which will be allowed to view results within the module; if the feature is disabled only users with edit rights will be able to view the results
- **Allow non admins the ability to purge form results** – select this option to allow the users with "view" rights to purge the form results
- **Allow non admins the ability to export form results** – select this option to allow the users with "view" rights to export the form results
- **Allow non admins the ability to manage form results** – select this option to allow the users with "view" rights to manage the form results
- **Display results directly within user-facing form display** – use this option to define the location of the results i.e. in the header or footer of the form; this is useful for instances of a module such as comments or ratings where you might want a user to be able to leave feedback that would be displayed directly on the user-facing page

7.51 Managing Form Completion Events

In order to start managing form completion events, choose option "Manage Questions/Settings" from the main menu and then click on the "Form Completion Events" icon.

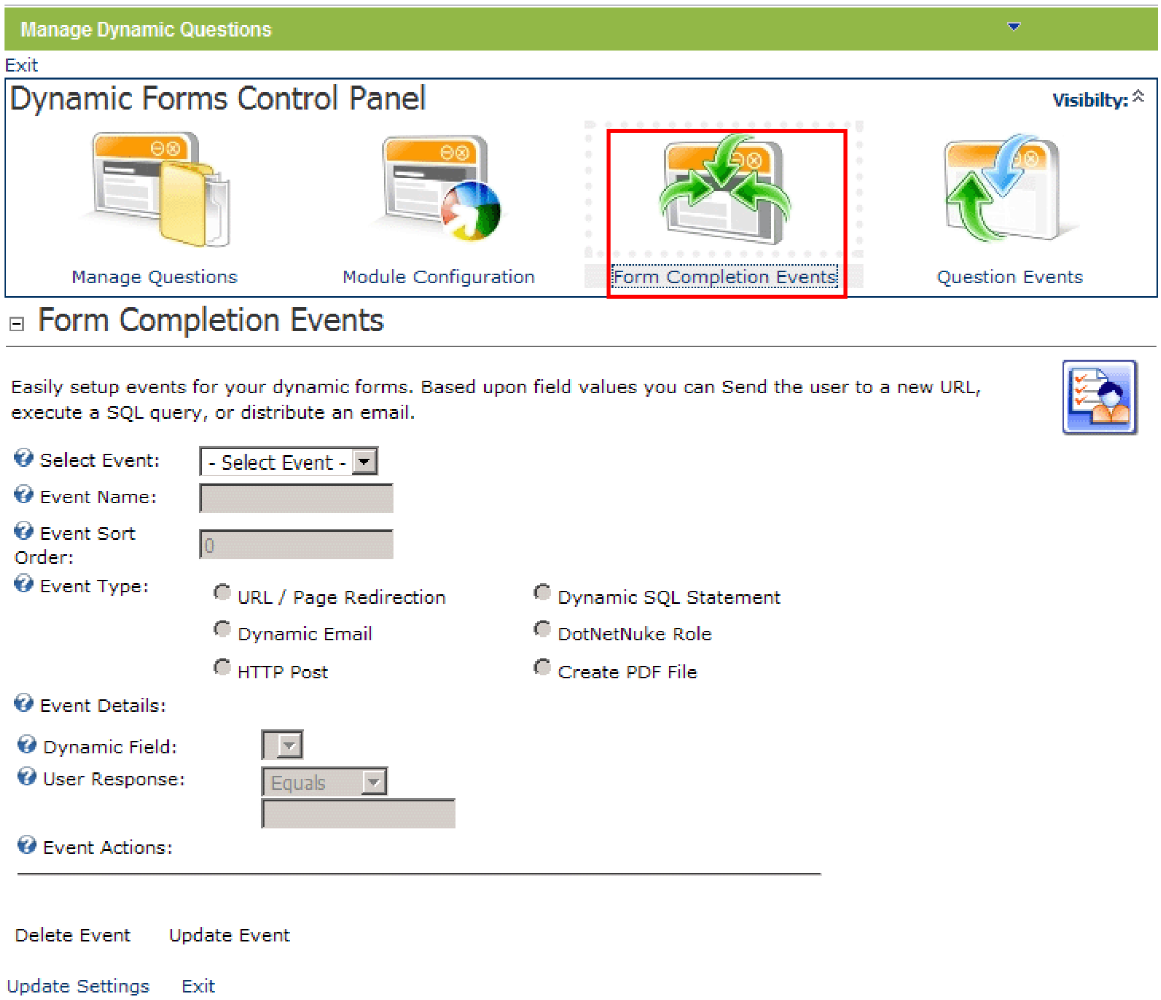


Figure 149: Managing events

The following options are available inside this screen:

- **New Event** – see section 7.52
- **Editing an existing event** – see section 7.52.7
- **Deleting an event** – see section 7.52.8

7.52 Creating a new event

In order to start creating a new event choose option new from the “Select Event” pull down menu and choose the desired event type.

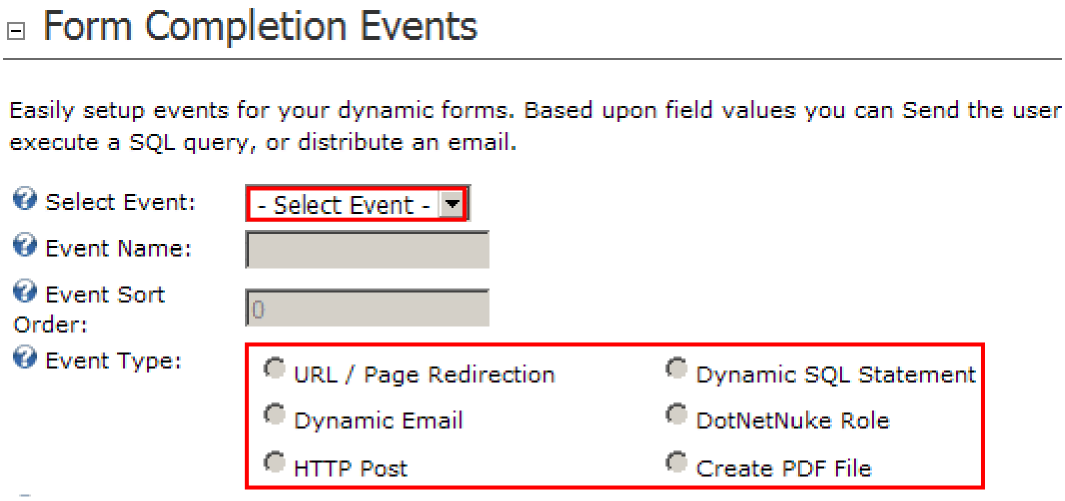


Figure 150: Creating a new event

The following are the available event types:

- **URL/Page Redirection** – choose this option if you want to redirect the user submitting the form based on the answers given in the form (see section 7.52.1)
- **Dynamic SQL Statement –** the dynamic sql event is for advanced users who wish to execute a SQL insert/update statement upon form submission. For example, if you wanted to insert form results into a log table you could execute a statement similar to this:

  Insert Into MyLogTable(FirstName, LastName, EmailAddress) Values($(MyFirstName), $(MyLastName), $(MyEmailAddress). – see section 7.52.2
- **Dynamic Email** – choose this option if you want to send an email either to the administrator or the user submitting the form based on the answers given in the form (see section 7.52.3)

  DotNetNuke® Role – the option for assigning a desired role on the system based on the response from the user (see section 7.52.5)
- **HTTP Post** – choose this option if you wish to post data from the form to another URL (**note**: you can post parts of the form or whole form to any URL you want i.e. web service such as SalesForce.com or virtually any form you can post to) – see section 7.52.4
- **Create PDF File** – the option for creating an event which will generate a PDF file – see section (7.52.6)

7.52.1 Example of the URL/Page Redirection Event

URL Page Redirection event is used for redirecting the user to a desired page based on the response given within the form.

For the purpose of this example a website offering the services for job seekers and employers is taken.

Since there are two types of users, you can create a dynamic question asking the user about the account type i.e. whether he is an employer or a job seeker (choosing option radio buttons as a question type).

Then you could create URL/Page Redirection event which will redirect the user to the page appropriate for his account type. In other words, you would redirect employers to the page containing information for employers and job seekers to their appropriate page.

In order to start creating URL/Page redirection event, choose option “New Event” from the “Select Event” menu and then select option “URL/Page Redirection”.

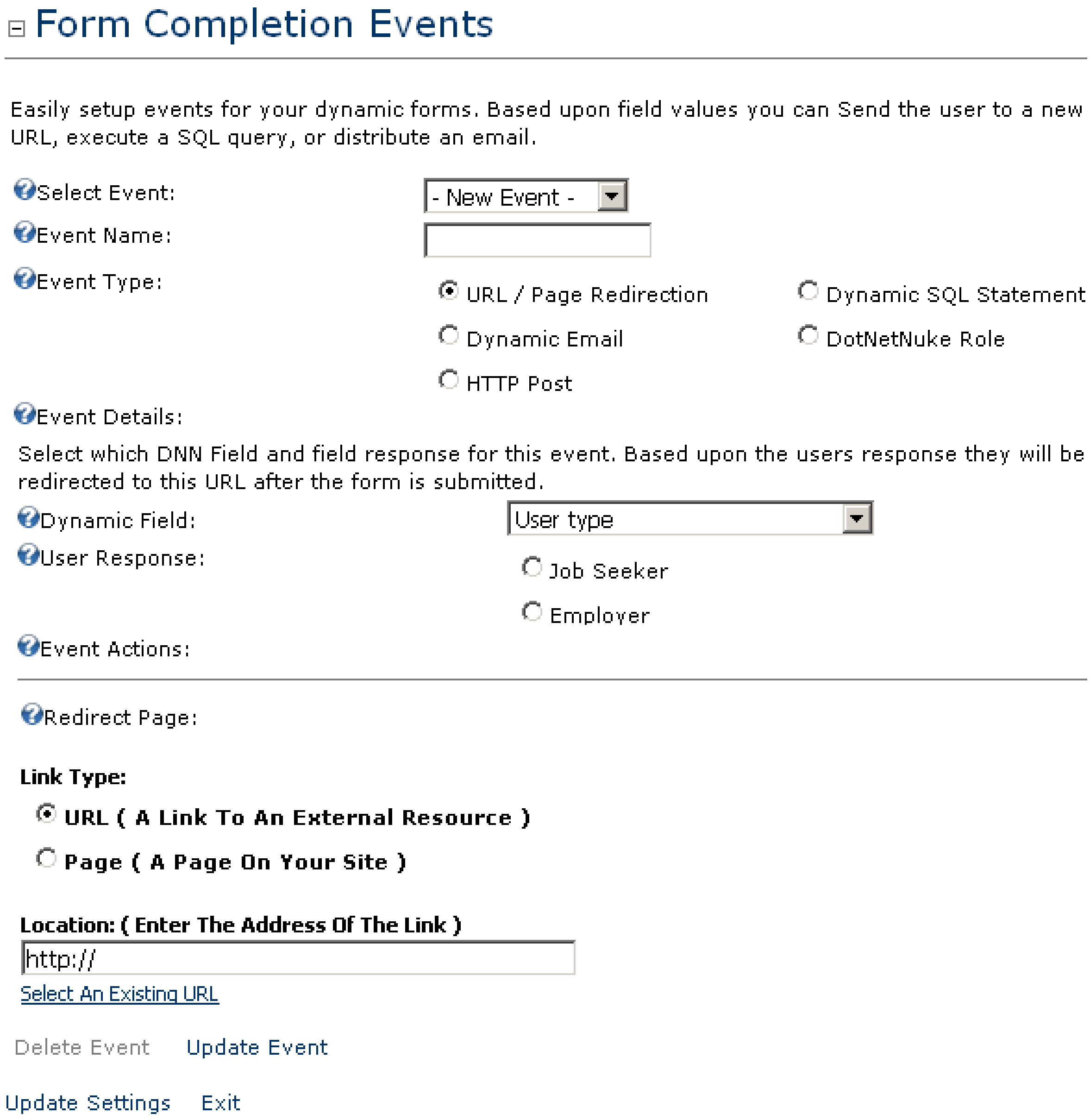


Figure 151: Creating URL/Page Redirection Event

The following are the parameters for defining the URL/Page Redirection event:

- **Dynamic Field** – choose the dynamic field you want to associate with this event (**e.g.** account type question asking the users to state if they are employers or job seekers)
- **User Response** – once you choose the desired dynamic field, the user response field will be refreshed with available options for choosing which user response will be associated to the event (**e.g.** “Job Seeker”); in this case the event will be applied to all users that chose “Job Seekers” as an account type
- **Event Actions** – this is the pull down menu for choosing the page on your website that the user will visit based on the given response (**e.g.** if the chooses “Job Seeker” as a response, he will be redirected to the page containing information for “Job Seekers”)

After setting these parameters click on the “Update Event” button and the new event will be created.

Note: you will need to create another event for the “Employer” which would redirect the users to the page containing information related to employers.

7.52.2 Example of the Dynamic SQL Statement event

In order to start creating the "Dynamic SQL Statement" event, choose option "New Event" from the "Select Event" menu and then select option "Dynamic SQL Statement" option.

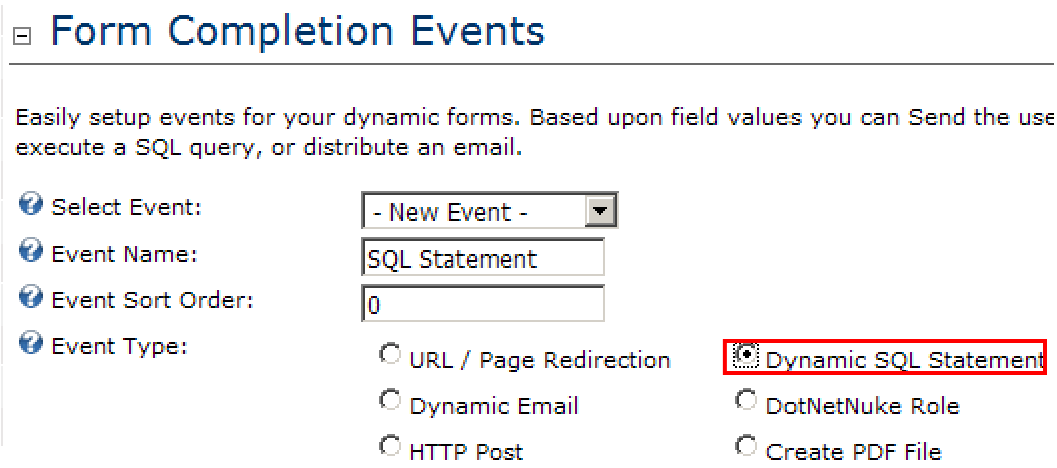


Figure 152: Creating a Dynamic SQL Statement event (step 1/2)

The screen will be refreshed containing the parameters for setting up the Dynamic SQL Statement event.

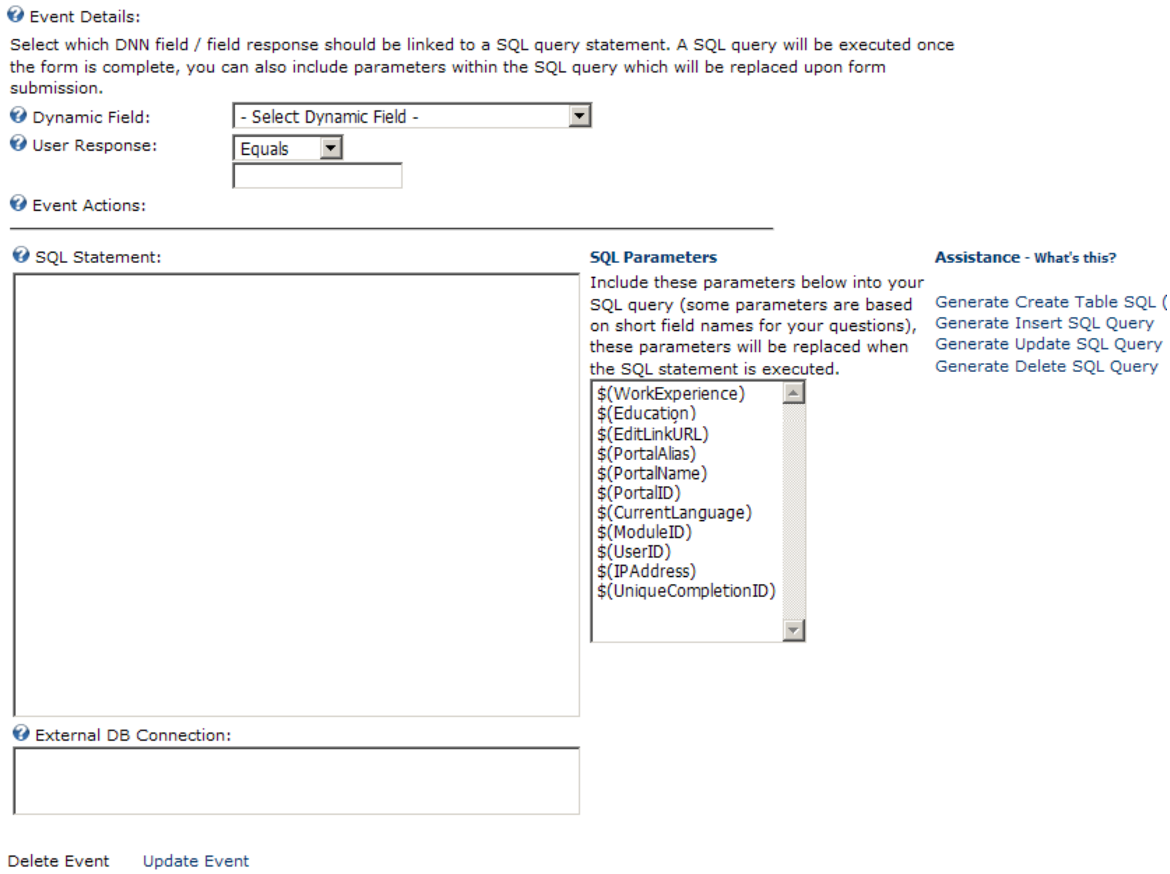


Figure 153: Creating a Dynamic SQL Statement event (step 2/2)

The following parameters and options are available:

- **Dynamic Field -** select the dynamic field that this event will based on
- **User Response -** select the response for this event that will cause the event to initialize

- **Event Actions** – this part of the screen is used for defining the event action and contains the following parameters:
    - o **SQL Statement** – the text area for entering the SQL statement that will be executed based on the users response or after the form results have been saved
    - o **SQL Parameters** – this is a list of SQL parameters which you can use in your SQL query (some parameters are based on short field names for your questions); to include the desired parameter **e.g.** First name, you can either double click it in the menu or drag and drop it inside the SQL Statement text area
    - o **Assistance links** - The auto generate SQL assistance links help to provide the functionality and quick assistance with building work-flow applications using Dynamic Forms. Dynamic Forms currently already saves data into relational database tables, and from those tables you can query the results/export the results to excel/view the results within the 'View Results' area of the module. The assistance has been added to provide further implementation assistance for users wanting to create Insert/Update/Delete implementations for a 'flat' table within your web site. The following assistance links are available:
        - **Generate Create Table SQL (One time only execution)** - this should be created after your form fields have been setup. A SQL script will be created which will generate a table based on the forms instance, and create a column within the table based on each form fields 'short field name'. This query should only be executed one time and should be executed under Host, SQL.
        - **Generate Insert SQL Query** - this should be created and utilized as a SQL Completion Event. You might consider creating a hidden field to represent 'Submit', 'Edit', 'Delete' functionality as described within the demonstration #14, and then only firing the insert statement for initial form submission.
        - **Generate Update SQL Query** - this should be created and utilized as a SQL Completion Event. You might consider creating a hidden field to represent 'Submit', 'Edit', 'Delete' functionality as described within the demonstration #14, and then only firing the update statement for editing or updating a form submission
        - **Generate Delete SQL Query** - this should be created and utilized as a SQL Completion Event. You might consider creating a hidden field to represent 'Submit', 'Edit', 'Delete' functionality as described within the demonstration #14, and then only firing the delete statement for deleting a form submission.
    - o **External DB Connection** – use this field in case you want to connect to a database outside of the standard DNN database connection

Note: As with all SQL Queries, you should always use caution and test against any possible SQL Injection attacks. Although Dynamic Forms does guard against SQL injection routines, you should always use stored procedures to guard against SQL Injection.

7.52.3 <u>Example of the Dynamic Email Event</u>

URL Page Redirection event is used for sending an email to the user or to you as an administrator based on the response given within the form.

For the purpose of this example a website offering the services for job seekers and employers is taken.

If this is the case you may want to create Dynamic Email event which would send an email to the user based on the response given in the form.

In order to start creating a Dynamic Email event, choose option “New Event” from the “Select Event” menu and then select option “Dynamic Email” (**note**: the screen is presented with two screenshots).

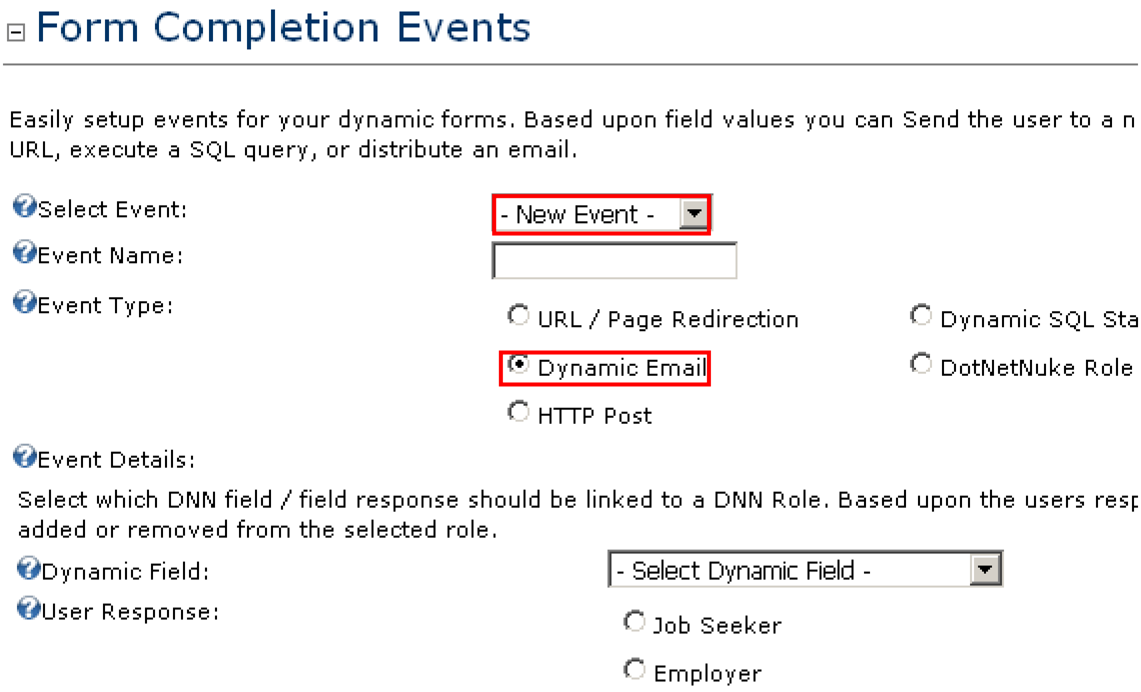


Figure 154: Creating a dynamic email event (screenshot 1/2)

The following parameters for defining a dynamic email event are available inside this part of the screen:

- **Select Event** – select option “New Event”
- **Event Name** – set the name for this email which will help you differentiate this event and its purpose (**e.g.** “Job Seeker Email”)
- **Event Type** – choose option “Dynamic Email”
- **Dynamic Field** – choose the dynamic field this event will be associated with (**e.g.** “Account Type”)
- **User Response** – once you choose the desired dynamic field, the user response field will be refreshed with available options for choosing which user response will be associated to the event (**e.g.** “Job Seeker”); in this case the event will be applied to all users that chose “Job Seekers” as an account type.
    - ***Note:*** New features were added to the 3.0 release to allow the response to either be ‘equal to’ and now ‘not equal to’ (previous versions always were set to only fire the event when the response was ‘equal to’.

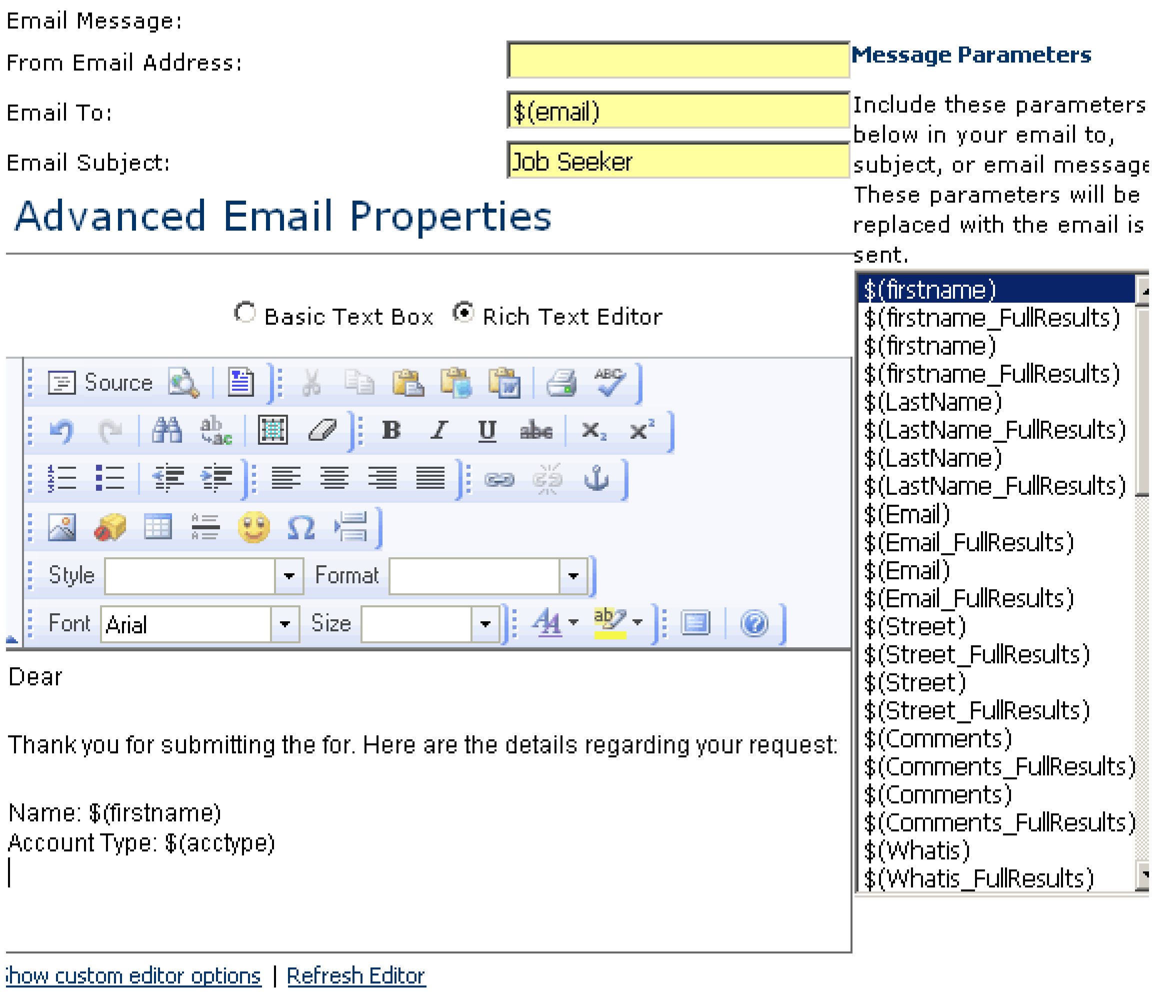


Figure 155: Creating a dynamic email event (screenshot 2/2)

The second part of the screen contains parameters for defining an email message that will be sent to the user:

- **From Email Address**– enter the from email address for this email event. The from email can either be static, such as 'host@yoursite.com' or it can be dynamic based on a fields response. To base the field on a users response you would use the short field name in between the parentheses, such as $(YourFromDynamicField).
    - o **Note:** If the from address is blank then the systems administrator address will be used as the from address
- **Email To** – enter the "short name" parameter for the email question; during the procedure of creating a dynamic question requesting email address form the users you are asked to set a short name for this parameter; in this example it is "email"; in order to send this email to the user submitting the form you need to enter this variable into the "Email To" input field (e.g. $(email) where "email" is the short name for the email address question and **$()** are the symbols telling the application that this is a variable)
    - o **Note:** alternatively you can enter your email if you want to receive an email notification once the user submits the form
- **Email Subject** – input field for entering email subject
- **Email Body** – text area for entering the body of the email message; this space can also be used for incorporating responses for all previously created dynamic questions (e.g. if you have created a question "account type" and given it a "acctype" short name, then enter **$(acctype)** inside the body of the email message to display this information to the user i.e. "job seeker")

- **File Attachments:** You can use the short field names for images and file attachments within the email just as you do with other parameters within the email body. For example $(MyFileUpload) or $(MyImage). The difference is that the file attachment is not actually attached, its references as the HTTP reference within the email to download the file. For example:
    - o Please download the file here $(MyFileUpload)
    - o To include it as a click use html such as this: Click here to download this file
    - o ***Tip:*** If you want to include the file attachments within the email as actual attachments then include the text “AttachFiles” within the email body. If this text is included within the email it will automatically be removed by the module and the file attachments will be added to the email.

Note: New tokens were added in version 2.5. These tokens reflect several new features added to the module*.*

- *$(ShortFieldName_FullResults)* – For each form field, a new token is available to display both the field label and the users’ response as one token. This token will only render results if there is a response from the user
- $(ShortFieldName_Text) – You can now retrieve text values as tokens for fields such as Combo Box, Radio Button, ListBox, and Check Box Groups. Previously you were only able to retrieve a question option value and not the text. For example if you specified a question option with the text of ‘California’ but a value of ‘CA’ you can use $(ShortFieldName) to retrieve ‘CA’ and $(ShortFieldName_Text) to render ‘California’.
- $(CalculateList_ShortFieldName) – You can now have a form ‘total’ items from a checkbox group or multi-selection listbox field types.
    - o To set values within the checkbox group/listbox fields you need to set each list items value to be VALUE01_Price. You can have up to 20 items with values in it.
    - o For example VALUE01_19.99, VALUE02_9.99, VALUE03_29.99, VALUE04_19.99, VALUE05_9.99).
    - o By using this feature you can create total fields such as 'Check all items you want to purchase' or 'Check all add-on's you want to add' and Dynamic Forms will total all of the checked or selected items for you with this token.
- $(IPAddress) – Including the $(IPAddress) token within the email will render the users IP Address
- $(UniqueCompletionID) – Include the unique completion ID to identify this individual form submission
- $(DateTime) – Include the current date and time of the submission
- AttachFiles - If you want to include the file attachments within the email as actual attachments then include the text “AttachFiles” within the email body. If this text is included within the email it will automatically be removed by the module and the file attachments will be added to the email. These are for file attachments using the File Upload field type. This is not necessary if you are setting up an attachment within the advanced completion email event settings.

After setting these parameters click on the “Update Event” link in order to complete the procedure of creating Dynamic Email event.

7.52.4 Setting up a HTTP post completion event

The HTTP post completion event is used in case you wish to post data from the form to another URL (**note**: you can post parts of the form or whole form to any URL you want i.e. web service such as SalesForce.com or virtually any form you can post to).

In order to start setting up this type of completion event, select “HTTP Post” within the “Event Type” part of the screen.

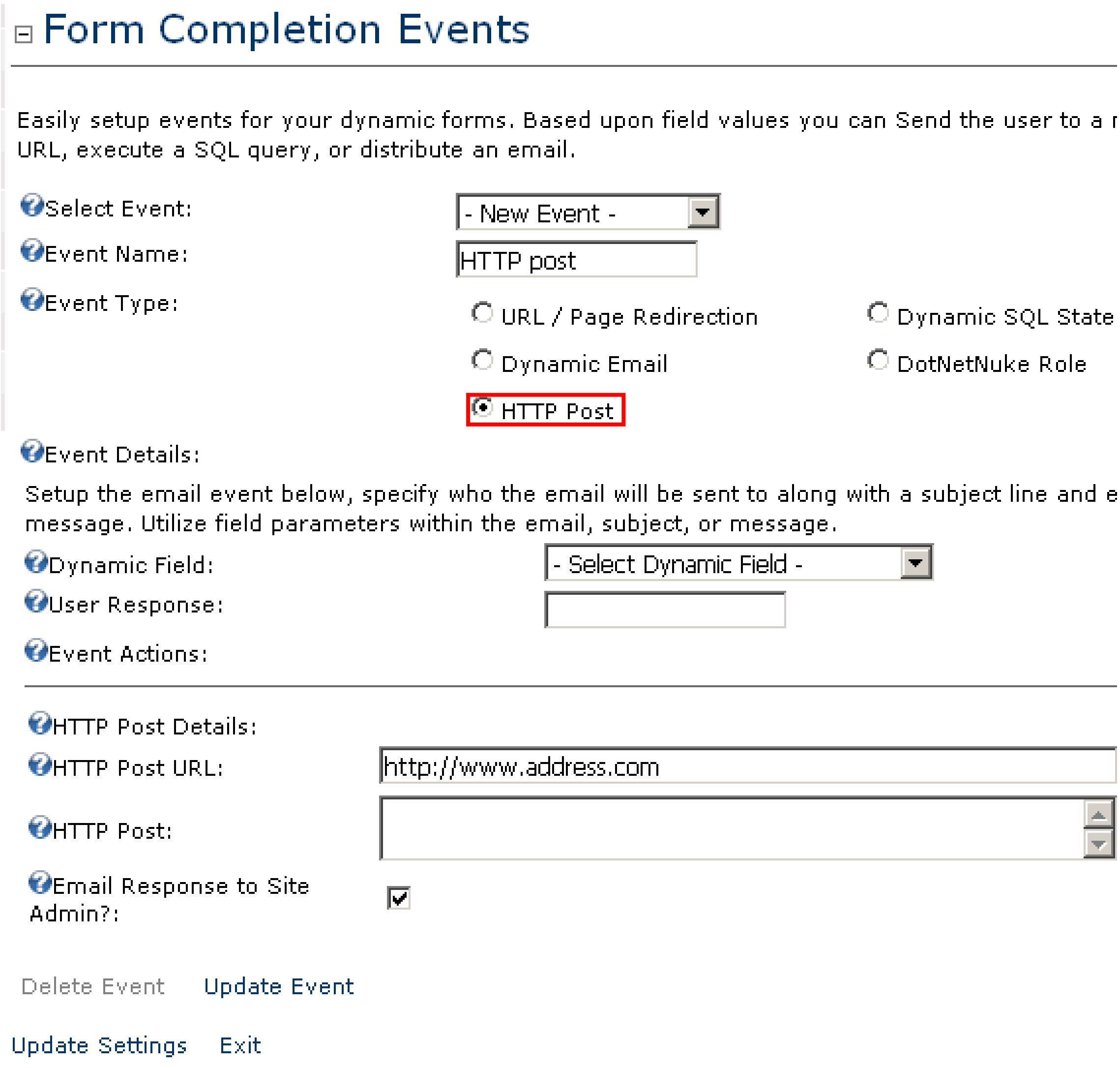


Figure 156: Setting up a HTTP post completion event

The following other parameters are available:

- **HTTP Post URL** – this is the actual URL which the form should post an HTTP Post to
- **HTTP Post** – this is the post which should be processed; the post should include a starting post variable followed by = and then the post response.
    - **For example**, a HTTP Post could be. Myname=Test&MyLastName=Test2&MyEmail=Test3
    - **Note**: a sample test post URL that you can post to is. This will render your post and return each parameter/response you are posting: http://authors.aspalliance.com/aldotnet/examples/posttest.asp
- **Email Response to Site Admin** – you might want a confirmation that the HTTP Post returned a result; if so you can choose to email the site administrator the results from the HTTP Post.

After setting the desired parameters, click on the “Update Event” link in order to save the changes.

7.52.5 Setting a DotNetNuke® Security Role based on the response

The DotNetNuke® role completion event is used in case you wish to assign a specific role to the user based on the response within the form.

In order to start setting up this type of completion event, select “DotNetNuke® Role” within the “Event Type” part of the screen.

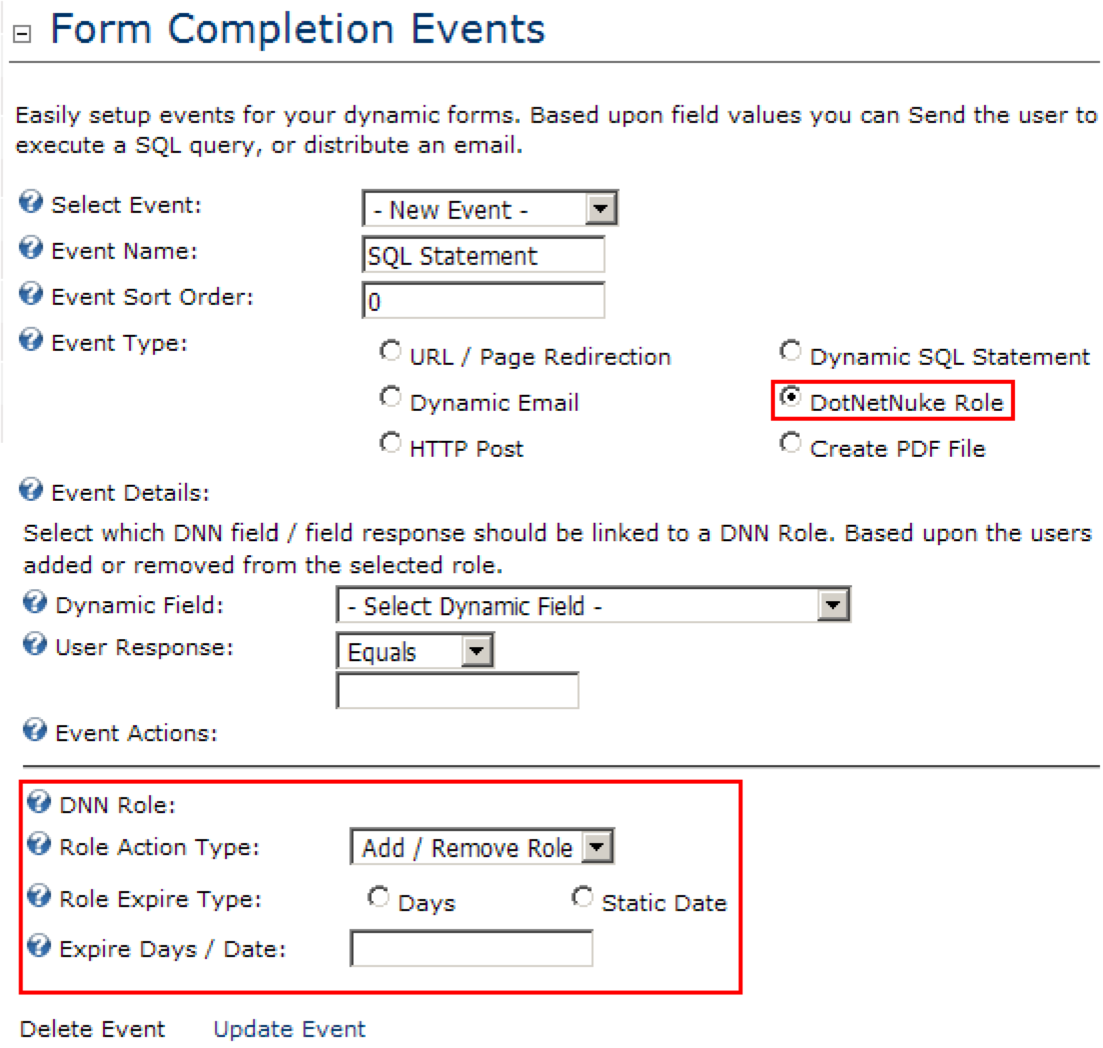


Figure 157: Setting up a DotNetNuke® completion event

The following parameters are available:

- **DNN® Role** – choose the role which should be assigned to the user
- **Role Action Type** – use this pull down menu to specify the type of action which should be performed to the user role; the following two options are available:
  - **Add / Remove** – choose this option to add a role based on the user response, or remove the user from the role if the user doesn’t match the response
  - **Add** – choose this option to add a role based on the user response
  - **Remove** – choose this option to remove the role from the user based on the response
    - **Note:** Prior to version 3.0 this completion event type always functioned in an add/remove action type. The user would be added to the security role if the response matched, and if the criteria did not match they would be removed from the security role. The updated enhancement allows you to setup many completion events for the same security role, without the risk of the user removing the security role etc…
- **Role Expire Type** – choose the type of role expiration; whether the role should expire in a set number of days (e.g. 20 days from today) or on a specific date in mm/dd/yyyy format (e.g. 01/01/2015). This field is used in combination with the field below where you should

enter the specific values according to the choice you have made here (either the number of days or a date). If this field is empty the role will be added without an expiration date.

- **Role Expire Days/Date** – set the number of days or the specific date this role will be assigned for (**e.g,** enter 45 days if you wish to assign the role for 45 days; after this period the user will automatically be demoted). Note: You can leave this field empty if you never want the role to expire.

Set the desired parameters and click on the "Update Event" link to save the changes. The screenshot below demonstrates the form as seen by the end users.

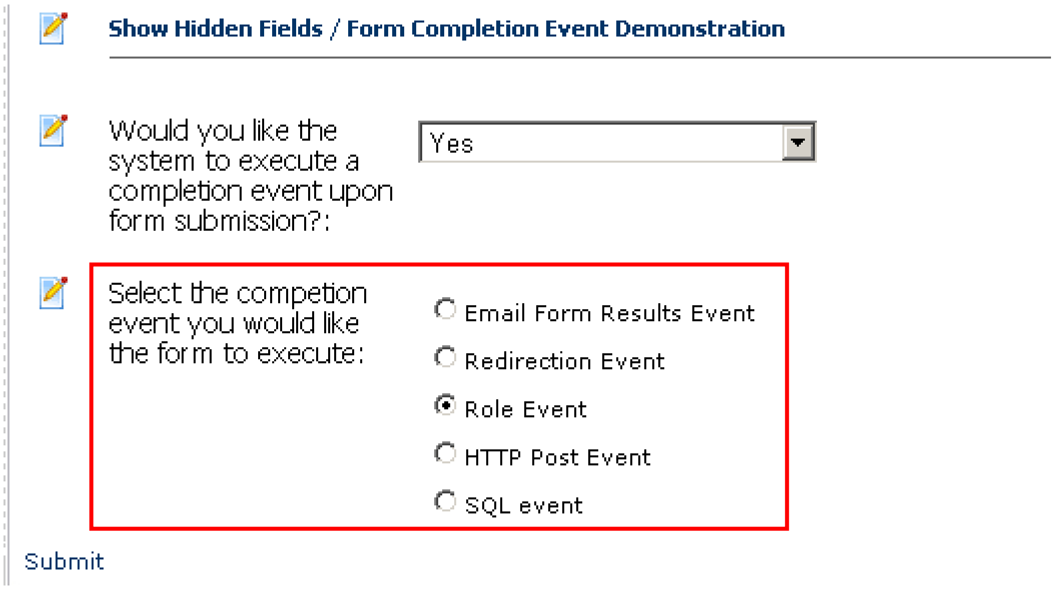


Figure 158: Example of assigning the role based on response

7.52.6 Creating the PDF file

The "Create PDF file" completion event is used in case you wish to create a PDF file which will contain the form contents.

In order to start setting up this type of completion event, select "Create PDFfile" within the "Event Type" part of the screen.

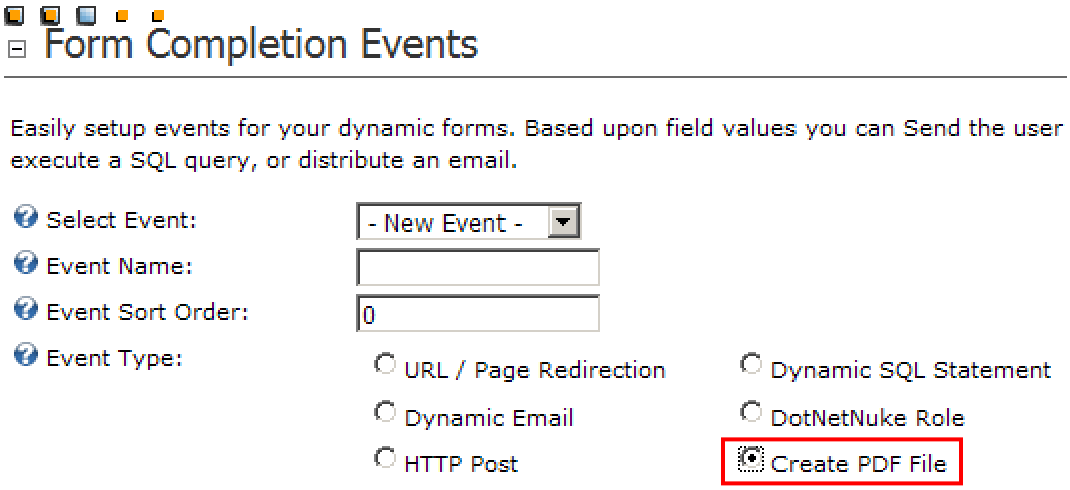


Figure 159: Creating a PDF file completion event

The following page will be displayed.

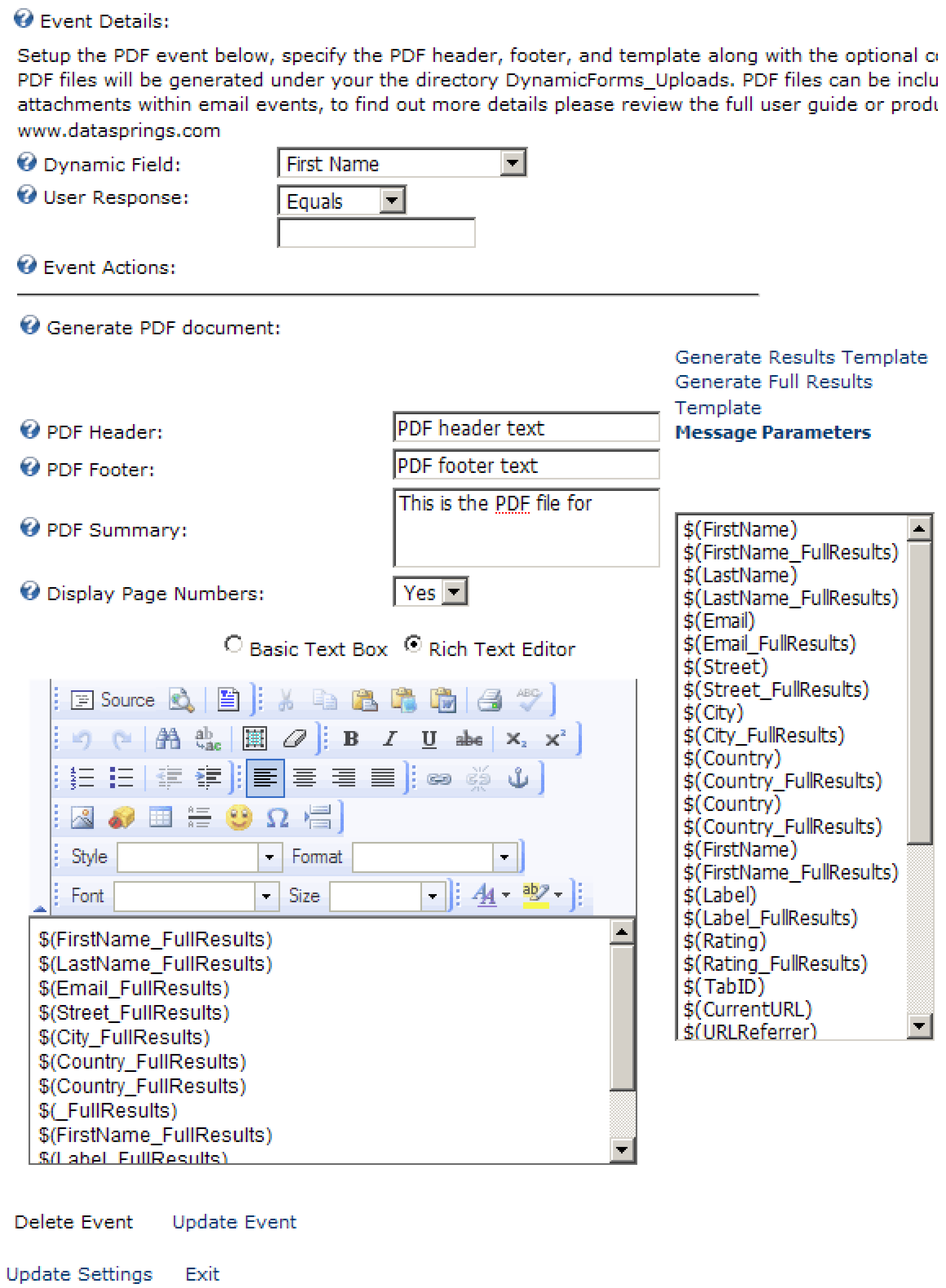


Figure 160: Creating the PDF

The following options and parameters are available:

- **PDF header/footer** – the fields for defining the PDF header and footer areas
- **PDF summary** – the field for entering the PDF summary
- **Display Page Numbers** – select whether or not you wish to display the page numbers within the PDF file

- **The body of the PDF** – the field for defining the body of the PDF file, i.e. the field which you will use to determine which parameters or results should be extracted and included in the PDF file
- **Generate Results Template** – use this option to automatically generate the results template
- **Generate Full Results Template**– use this option to automatically generate the results template
- **Message Parameters** – use the message parameters to create a custom set of parameters which should be displayed within the PDF file

Important note: PDF completion events can be used as an attached file within email completion events. You must setup the PDF completion event to have a sort order that is lower (i.e 5) then the email completion event (i.e. 10). Then within the email completion event you must reference the PDF event as a token (you should see an additional token in the listbox of available tokens). The tokens are $(PDFLink:EventName) and $(PDFAttach:EventName). You should replace 'EventName' with the name of the actual PDF completion event.

Note: For a detailed demo of this functionality please review this link:

http://www.datasprings.com/Products/DNNModules/DynamicForms/DynamicFormsDemo22SurveyGridviewPDF/tabid/940/language/en-US/Default.aspx

7.52.7 Editing an existing event

In order to edit an existing event, choose the desired event from the "Select Event" pull down menu.

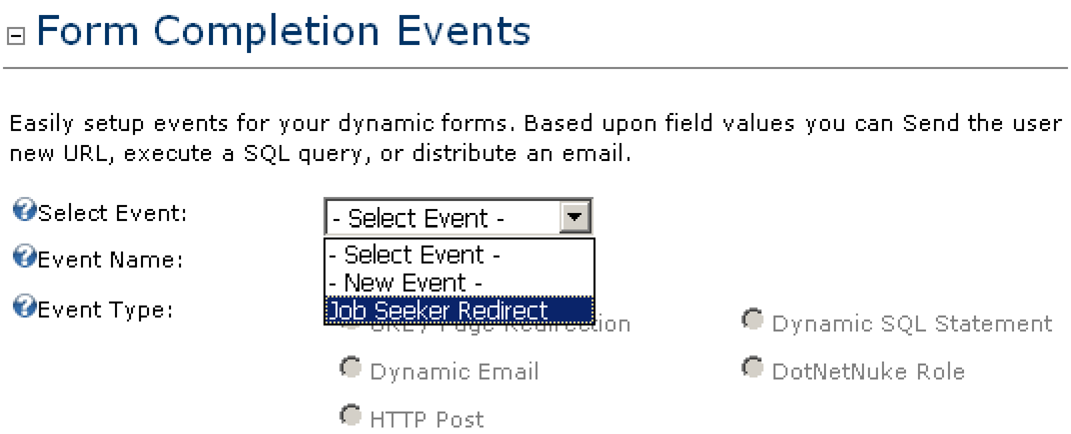


Figure 161: Editing an existing event

The page will be refreshed and the parameters set for the chosen event will be displayed. Change the desired values and click on the "Update Event" link in order to save the changes.

7.52.8 Deleting an event

In order to delete an event, choose the desired event from the "Select Event" pull down menu.

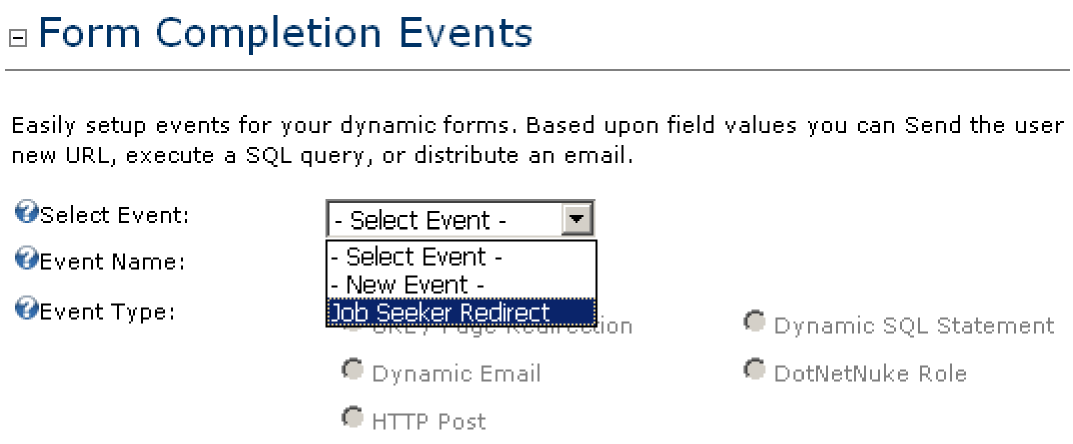


Figure 162: Deleting an event

The page will be refreshed and the option “Delete Event” will be available. Click on this link to delete the chosen event.

7.53 Managing Question Events

In order to start managing question events, choose option “Manage Settings/Questions” from the main menu and then click on the “Question Events” icon.

Figure 163: Managing Question Events

The following parameters are available inside this screen:

- **Question Event** – pull down menu for choosing the editing the desired question event or creating a new one
- **Event Name** – input field for entering the name of the new question event
- **Is disabled?** – select this option to disable the event; you can use the same option to enable it subsequently
- **Event Type** – options for choosing the type of the question event
    - **Hide question based on another question’s response** – see section 7.53.1
    - **Set Question Default based on another question response** – see section 7.53.2
    - **Display hidden question based on another question’s response** – see section 7.53.3
    - **Disable question based on another question’s response** – see section 7.53.4
    - **Execute Form Submission -** see section 7.53.5

- **Initiation / Postback Question** – select the desired postback question; it will be the specific response to this question that will affect displaying or hiding another question ("Affected Question" option)
- **Execute Event for any response** – select this option in case the desired event (hiding or displaying) should be executed no matter the response given to the postback question
- **Initiation Response** – pull down menu for choosing the initiation response, i.e. the response that will trigger the event (showing or displaying the affected question)
- **Affected Question** – pull down menu for choosing the question that will be affected by the event; in other words which question will be displayed or shown based on the response given for the postback question
- **Affected Question New Value -** If you have previously selected the option to change the value of a field based on the response to another field you can set the new value here.

7.53.1 Hiding a question based on the response

This type of event is used in case you want to hide a question after the user has given certain response to a given question.

For the purpose of explaining this feature, the example of asking the user about the type of PC has been taken.

The assumption is that most of the users filling the form will have a laptop and that you want to acquire further information about their preferred laptop accessory (either a bag or a backpack).

In case the user does own a laptop, he will select "Laptop" from the first question (which is a **postback question** in our example). Then this user can move on and supply information about the preferred accessory.

However, if the user has a "Desktop" PC, he will choose the "Desktop" option from the pull down menu (which is **initiation response**) thus making the question about the preferred laptop unnecessary. That question will be automatically hidden from the user (**affected question** in our case). Read below to find detailed instructions on how to define this type of an event.

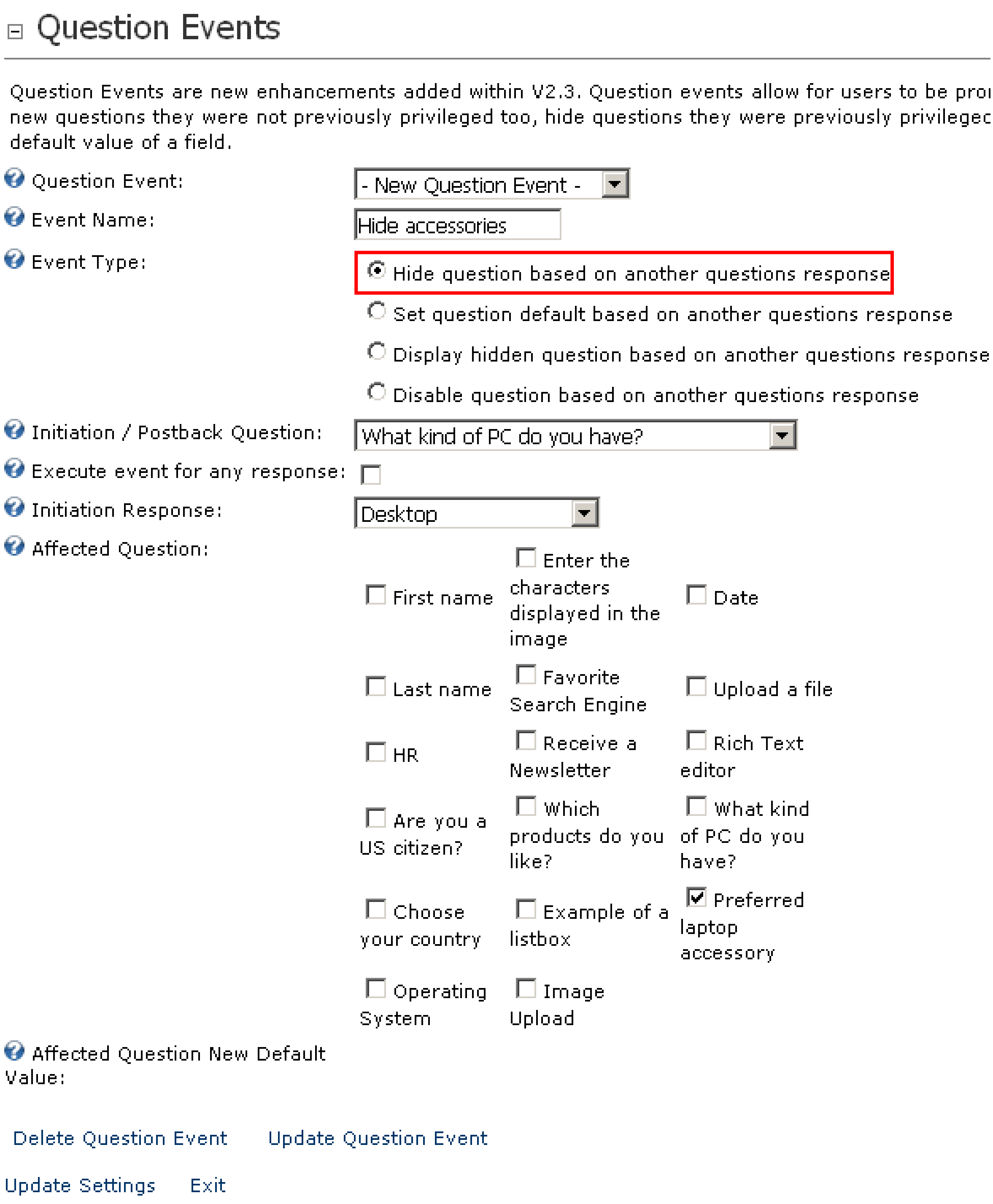


Figure 164: Hiding a question based on the response

- **Question Event** – choose option “New Question Event”
- **Event Name** – enter the name of the event (**e.g.** “Hide accessories”)
- **Event Type** – choose option “**Hide question based on another question’s response**”
- **Initiation / Postback Question** – select the desired postback question; in this case the postback question would be “**What kind of a PC do you have?**”
- **Initiation Response** – pull down menu for choosing the initiation response; in our example the initiation response would be “**Desktop**” which would cause the question about the accessories to be hidden
- **Affected Question** – pull down menu for choosing the question that will be affected by the event; in our example the affected question would be “**Preferred laptop accessory**”

After setting the desired parameters, click on the "Update Question Event" link and the new event will be created.

The screenshot below demonstrates this question event as seen by the end user.

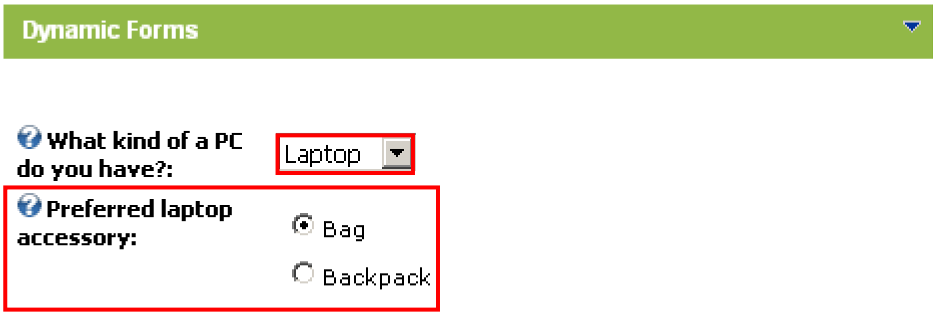


Figure 165: Example of this event as seen by the end user

The default value for the first question has been set to “Laptop”. Once the user chooses “Desktop”, the “Preferred laptop accessory” question will disappear.

7.53.2 Setting the question default based on the response

This type of event is used in case you want to set the default value of another field based on response from a previous field. For example, if the user check’s a box (initiation or postback question) you might want to set the default value to another field to be ‘Yes’.

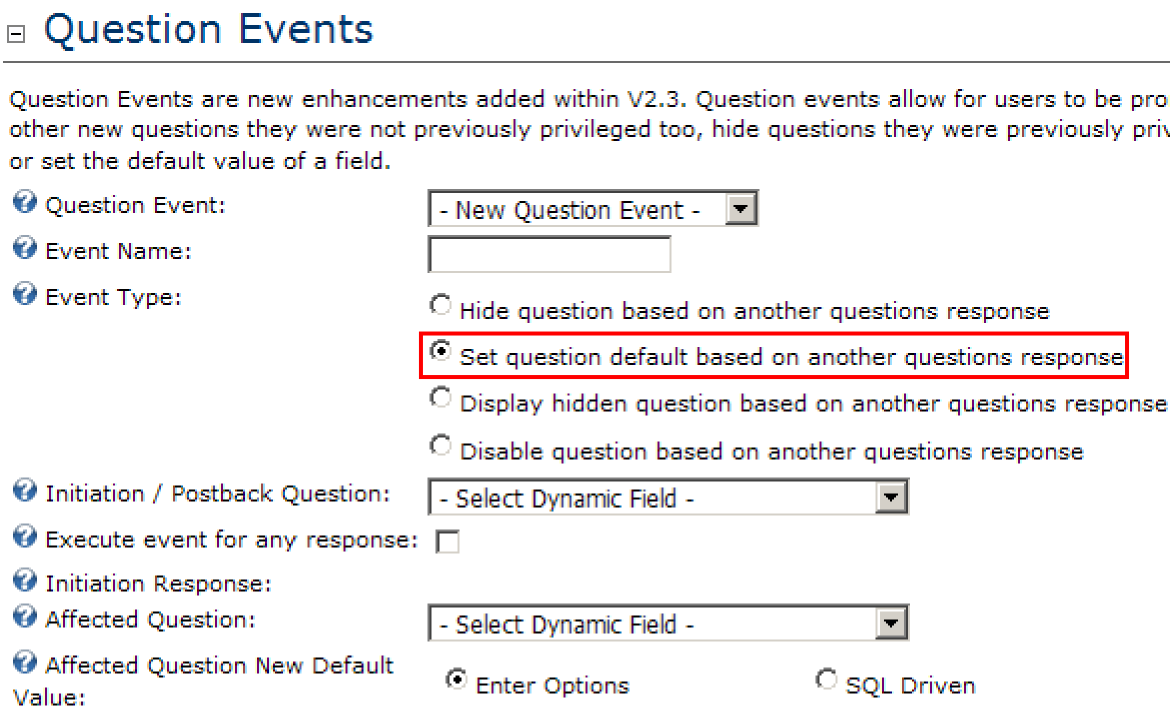


Figure 166: Setting the question default based on the response

- **Question Event** – choose option “New Question Event”
- **Event Name** – enter the name of the event
- **Event Type** – choose option “**Set question default based on another question’s response**”
- **Initiation / Postback Question** – select the desired postback question

- **Execute event for any response** – select this checkbox in case you wish to trigger the event whatever the response
- **Initiation Response** – pull down menu for choosing the initiation response; in our example the initiation response would be “**Yes**” which would cause the affected question (**PDA manufacturer**) to be displayed to this user
- **Affected Question** – pull down menu for choosing the question that will be affected by the event
- **Affected Question New Default Value** - if you selected the option to change the value of a field based on another field you can set the new value here by using any of the two following options:
    - **Enter options –** Enter the new value the field should render when the initiation response has been triggered.
    - **SQL driven -** Enter a query that should return a single row/column with the column name ‘DefaultValue’. Please note, all queries should use stored procedures to minimize the risk of SQL injection (especially within question events if you are referencing other field tokens).

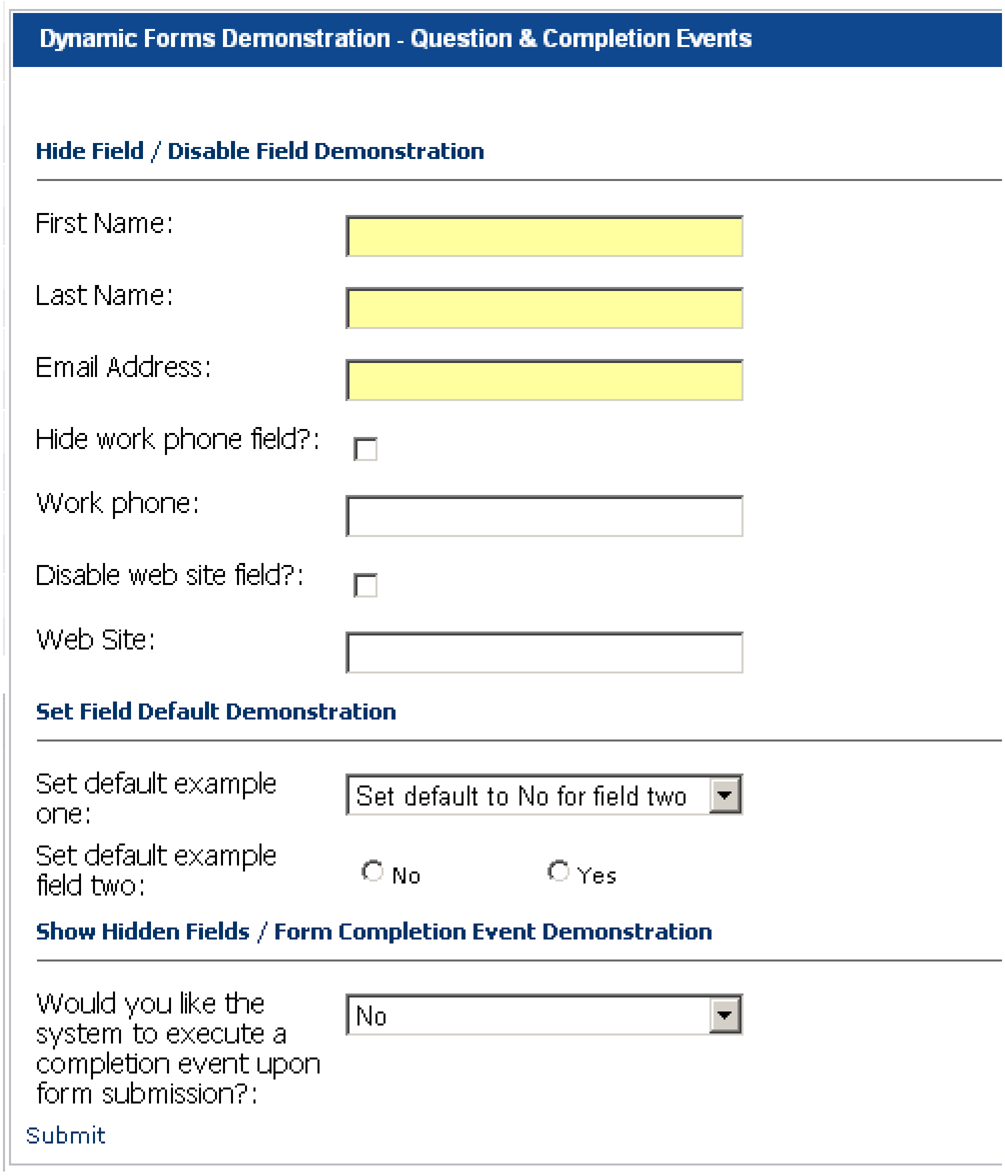


Figure 167: Example of setting the question default based on the response

Review Demonstration of Default Value Question Event

7.53.3 Displaying hidden question based on a response

This type of event is used in case you want to display a question that has been hidden from the user the user has given certain response to a given question.

For the purpose of explaining this feature, the example of asking the user about whether he has a PDA device or not has been taken.

Once the user supplies the answer “Yes” (**initiation response** in our case), the question about the PDA manufacturer will be displayed to the user so that he could choose the manufacturer of his PDA device. Read below to find detailed instructions on how to define this type of an event.

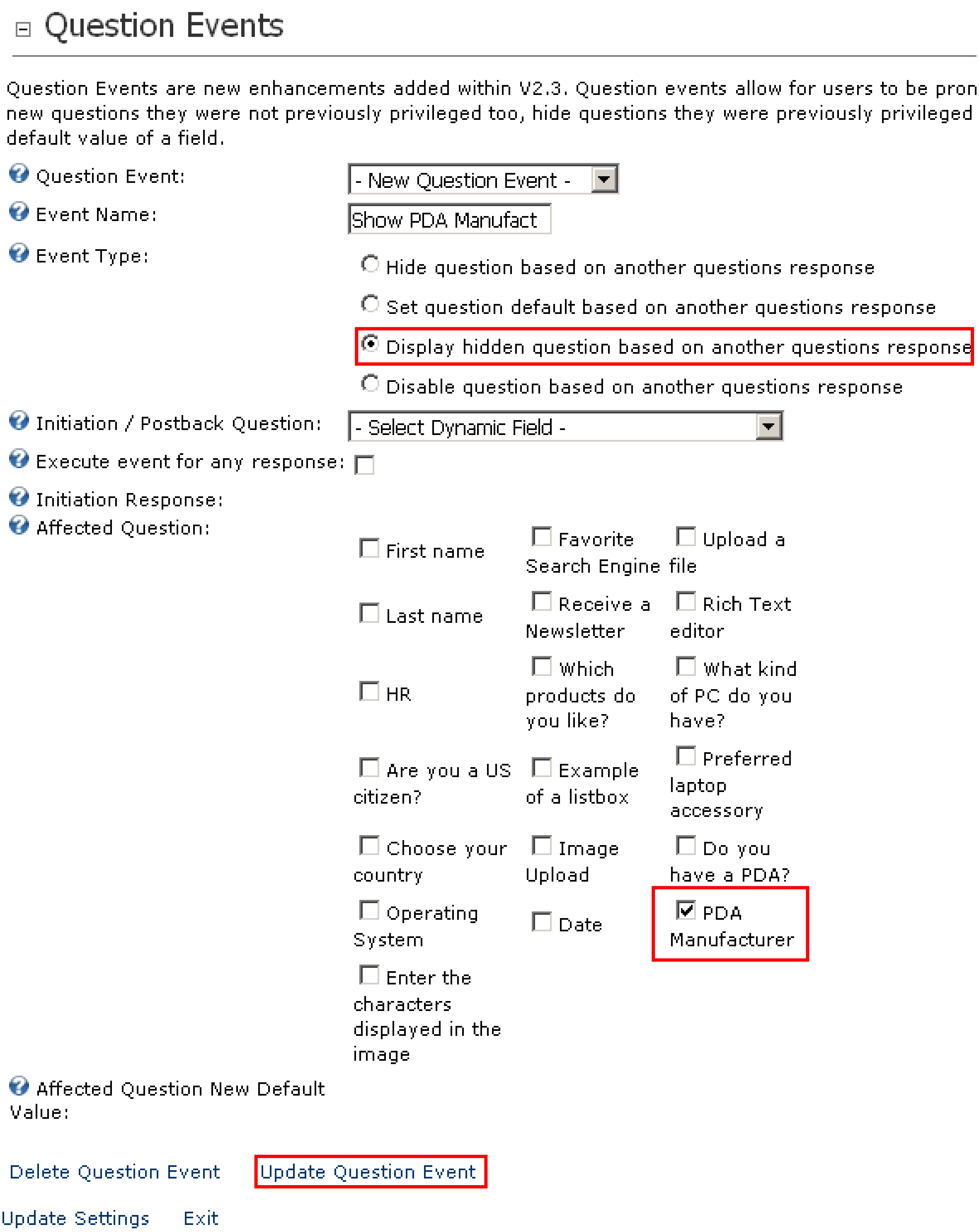


Figure 168: Displaying hidden question based on a response

- **Question Event** – choose option “New Question Event”
- **Event Name** – enter the name of the event (**e.g.** “Show PDA manufacturer”)
- **Event Type** – choose option “**Display hidden question based on another question’s response**”
- **Initiation / Postback Question** – select the desired postback question; in this case the postback question would be “**Do you have a PDA?**”
- **Initiation Response** – pull down menu for choosing the initiation response; in our example the initiation response would be “**Yes**” which would cause the affected question (**PDA manufacturer**) to be displayed to this user
- **Affected Question** – pull down menu for choosing the question that will be affected by the event; in our example the affected question would be “**PDA Manufacturer**”

After setting the desired parameters, click on the "Update Question Event” link and the new event will be created. The screenshot below demonstrates this question event as seen by the end user.

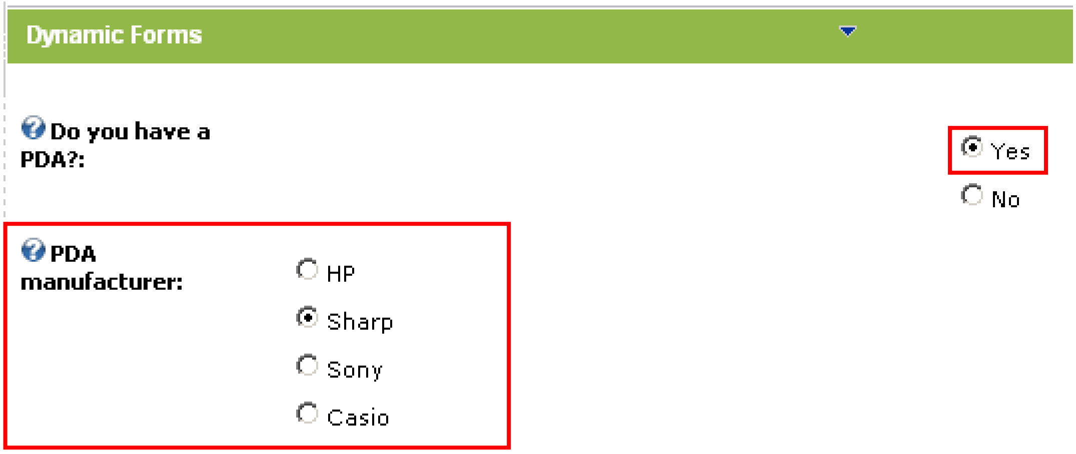


Figure 169: Example of the event question as seen by the end user

The “PDA manufacturer” question is hidden until the user answers “Yes” to the “Do you have a PDA” question.

Important note: in order for the affected question (**PDA manufacturer**) to be hidden until the user supplies the proper initiation response, you need to select the “**Hide until forced visible by question event**” option within the “Advanced Field Options” page.

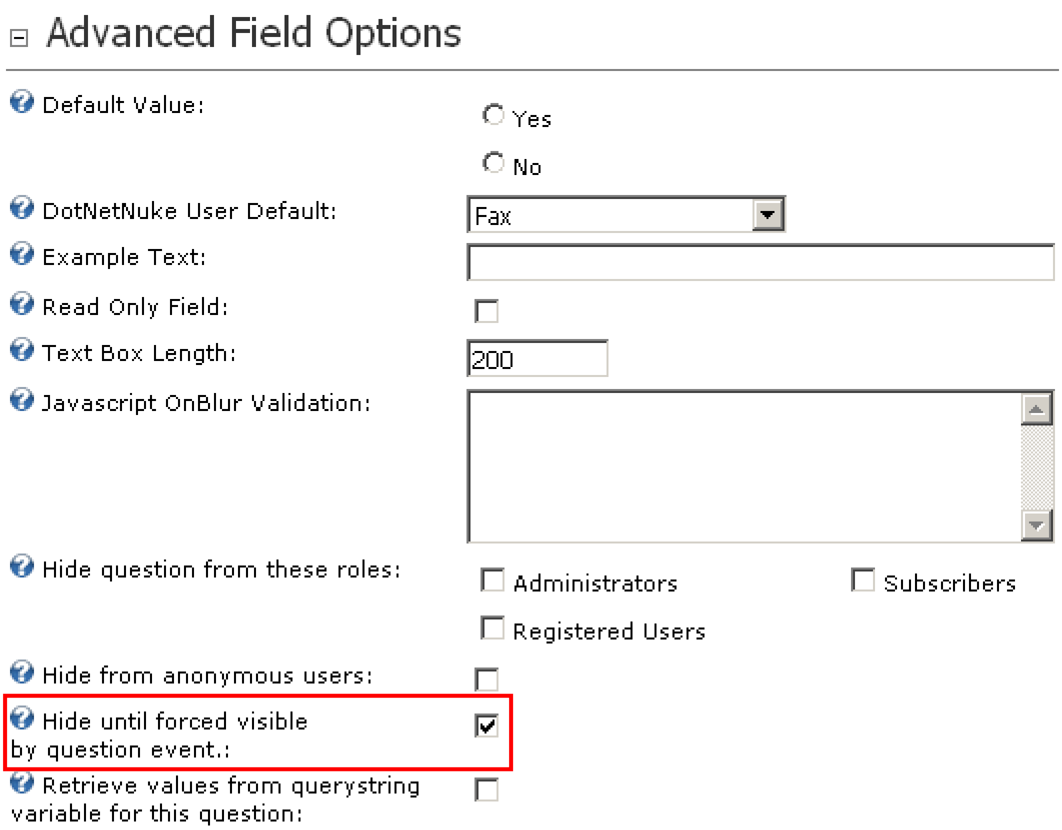


Figure 170: Editing "Advanced Field Options"

7.53.4 Disabling a question based on another question's response

This type of event is used in case you want to disable a certain question based on the response for another question. An example for using this option can be a situation where you want to differentiate users from US and those from the rest of the world.

In that case you could create a radio buttons element asking the user whether he is a US citizen or not. Based on the response (YES), you can determine to disable the pull down menu for choosing the country (containing all other countries but US).

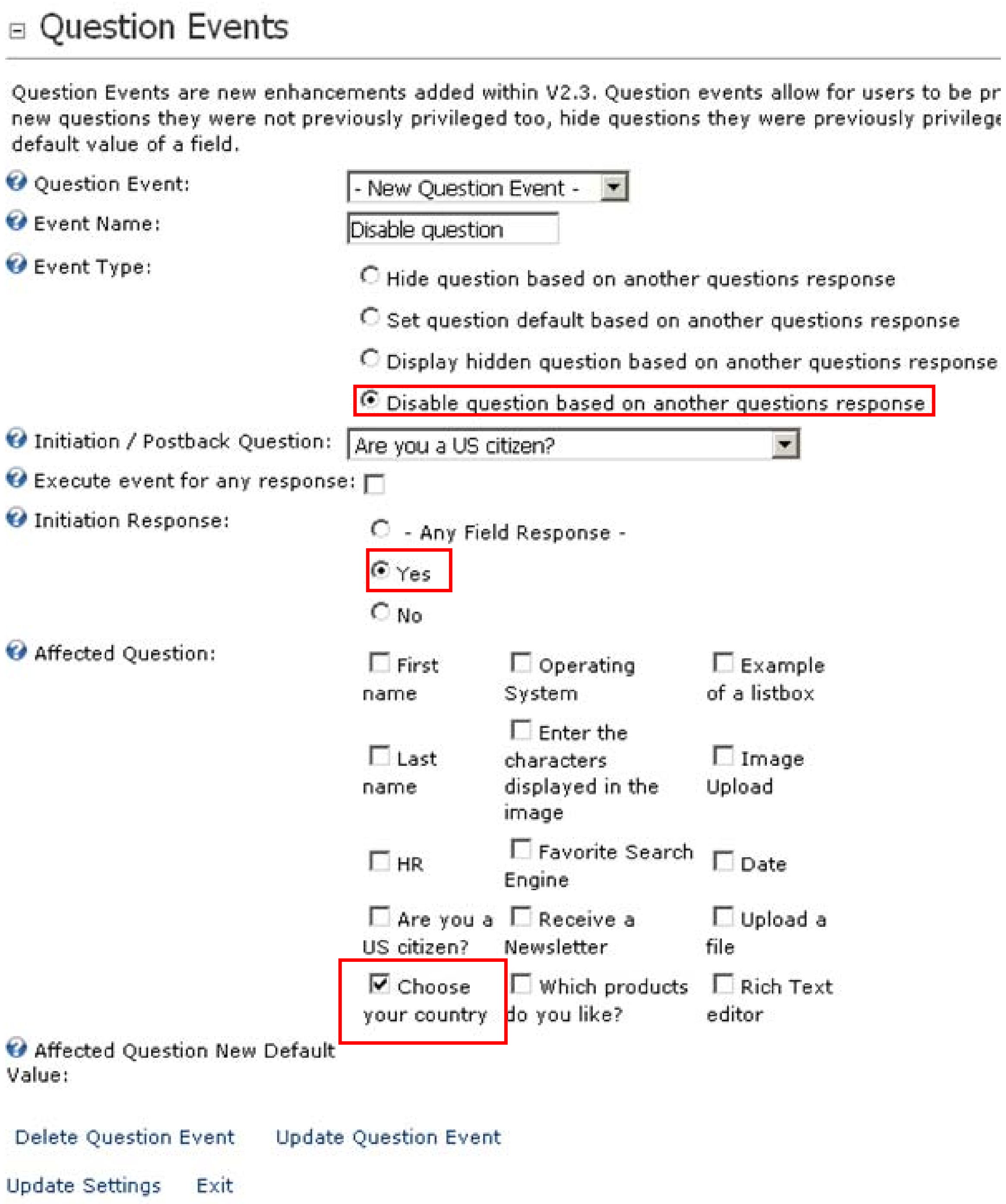


Figure 171: Disabling a question

The screen shots below demonstrate hiding of the field based on the response from the user.

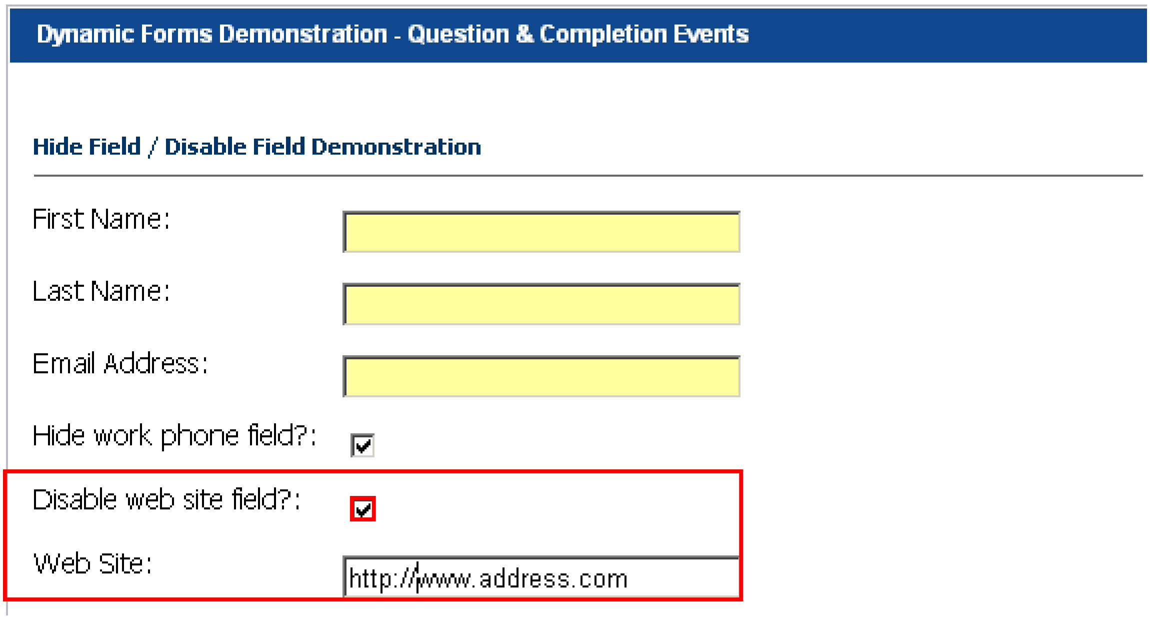


Figure 172: Example of disabling a question based on the response (step 1/2)

Once the user selects the “Disable web site field”, the screen will be refreshed and the “Web Site” field will be disabled.

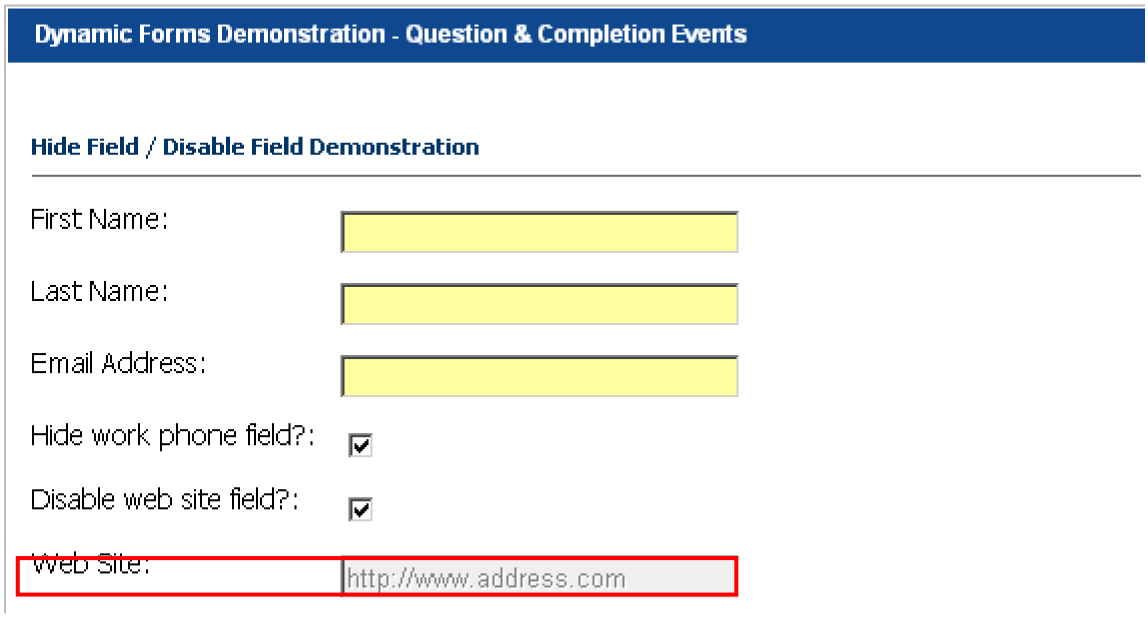


Figure 173: Example of disabling a question based on the response (step 2/2)

7.53.5 Executing the form submission

This question event is used for automatically submitting the form upon firing. In case the conditions are met, the form will be submitted instantly.

In order to create this question event, choose option “New Question Event” From the “Question Event” menu.

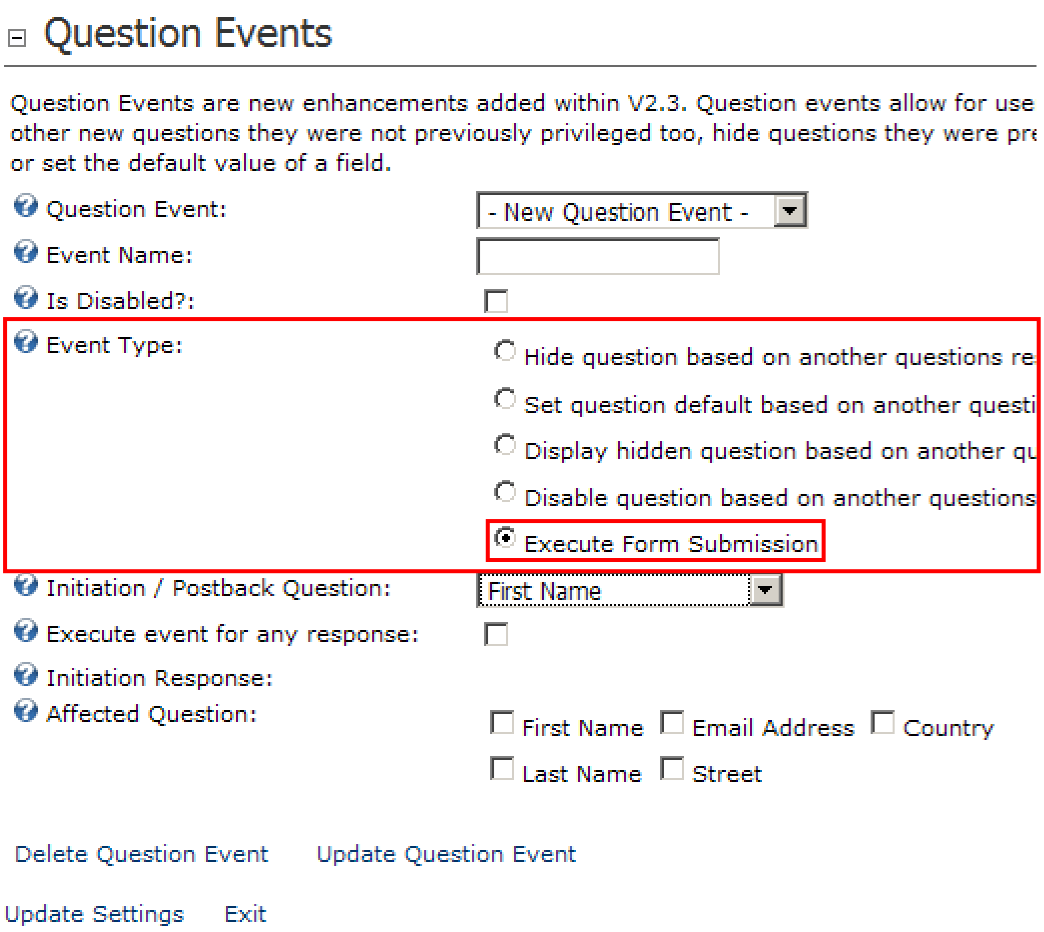


Figure 174: Executing the form submission

For the “Event Type” choose “Execute Form Submission” and then define the rest of the criteria.

7.53.6 Editing a question event

In order to edit a question event, choose the desired question event from the pull down menu.

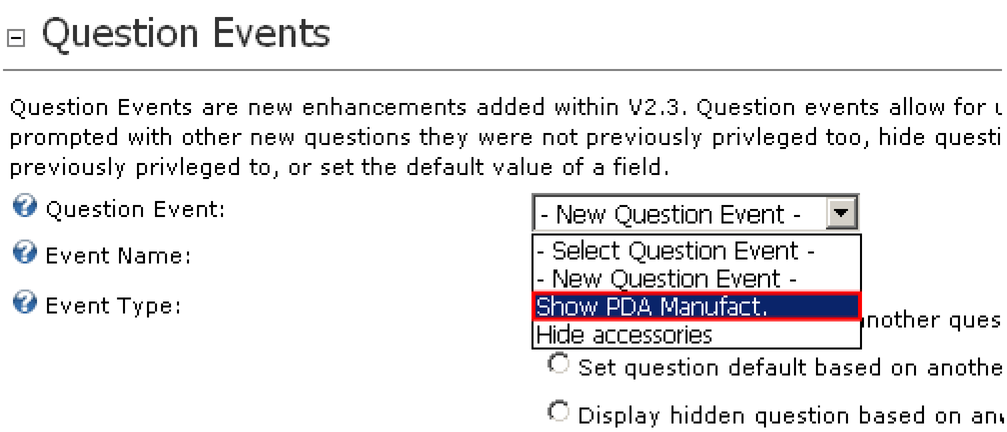


Figure 175: Editing a question event

The screen will be refreshed containing the parameters of the chosen question event where you can make the desired changes.

7.53.7 Deleting a question event

In order to delete a question event, choose the desired question event from the pull down menu.

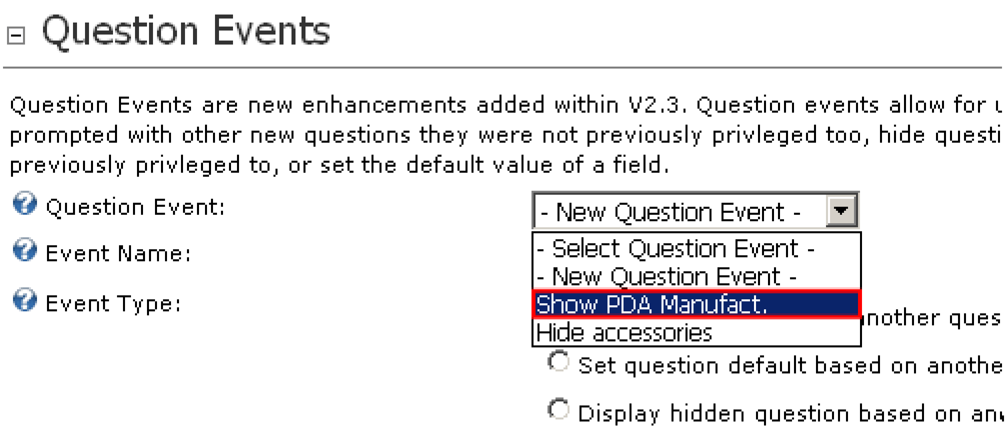


Figure 176: Deleting the question event (step 1/2)

The following screen will be displayed.

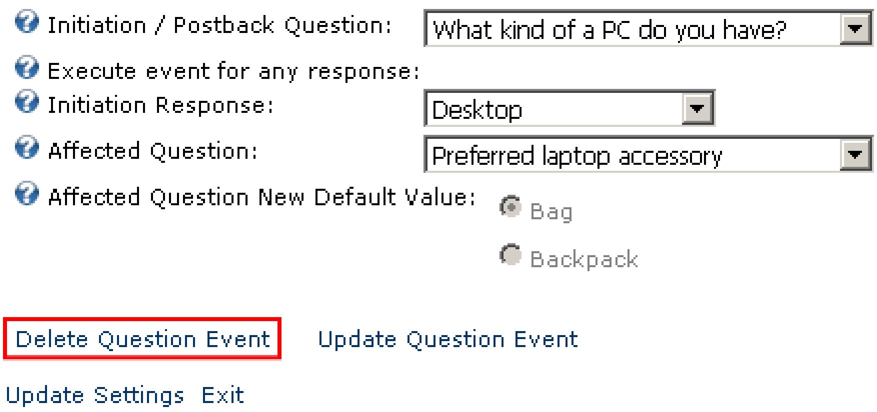


Figure 177: Deleting the question event (step 2/2)

Click on the “Delete Question Event” link and the selected question event will be deleted.

8 VIEWING FORM RESULTS

Note: the users with "view" (non-admin role) rights can also be allowed to view, edit, manage templates, export, and purge form results. For further information about this functionality, see section 7.50

In order to view the results of your form choose option "View Form Results" from the "Advanced Features" submenu.

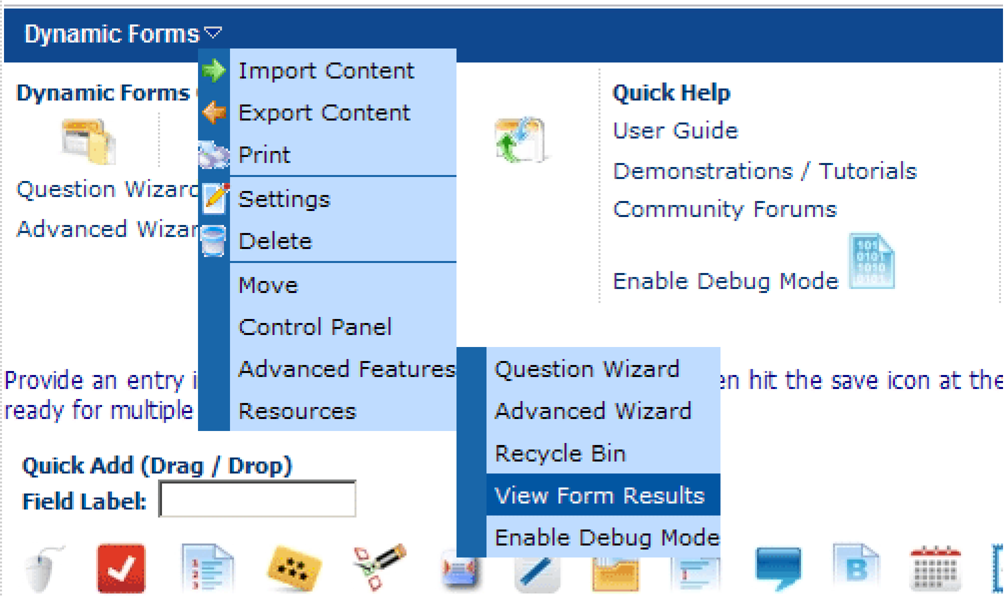


Figure 178: Viewing form results (step 1/2)

The screen containing summary information about users filling the form will be displayed.

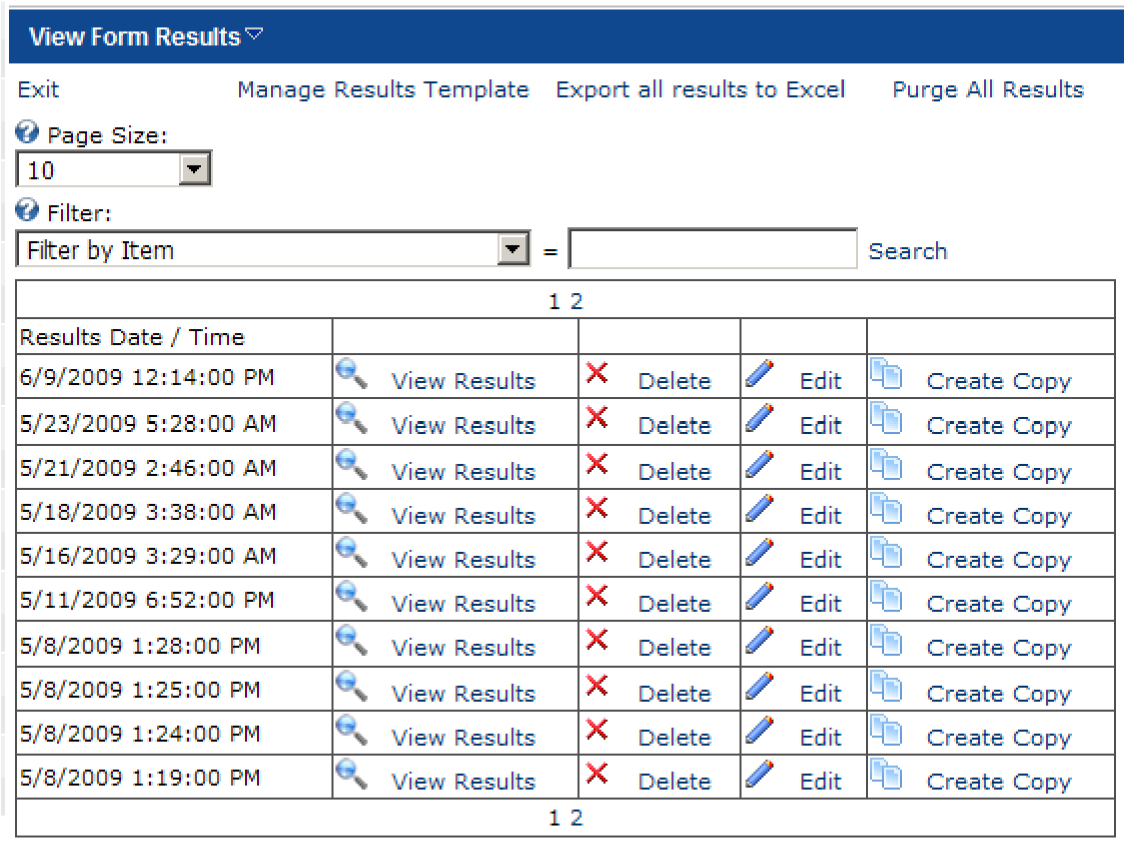


Figure 179: Viewing form results (step 2/2)

The following options are available inside this screen:

- **Manage Results Template** – the option for managing the results template (see section 8.2)
- **Export All Results to Excel** – click on this link to export all form results to an excel file (see section 8.7)
- **Purge All Results** – the option for purging the form results (see section 8.1)
- **Page size** – set the desired number of items per page
- **Filter** – select the field you wish to filter the results for and/or enter the keyword for performing the search
- **View Results** – click on this link to view detailed results of the form (see section 8.1)
- **Delete** – click on this link to delete the form result (see section 8.4)
- **Edit** – click on this link to edit the form result (see section 8.5)
- **Create Copy** – click on this link to create a copy of the form result (see section 8.6)
- **Export all results to Excel** – option for exporting results to Excel (see section 8.7)

8.1 Purging the form results

In order to purge i.e. delete all form results click the “Purge All Results” link.

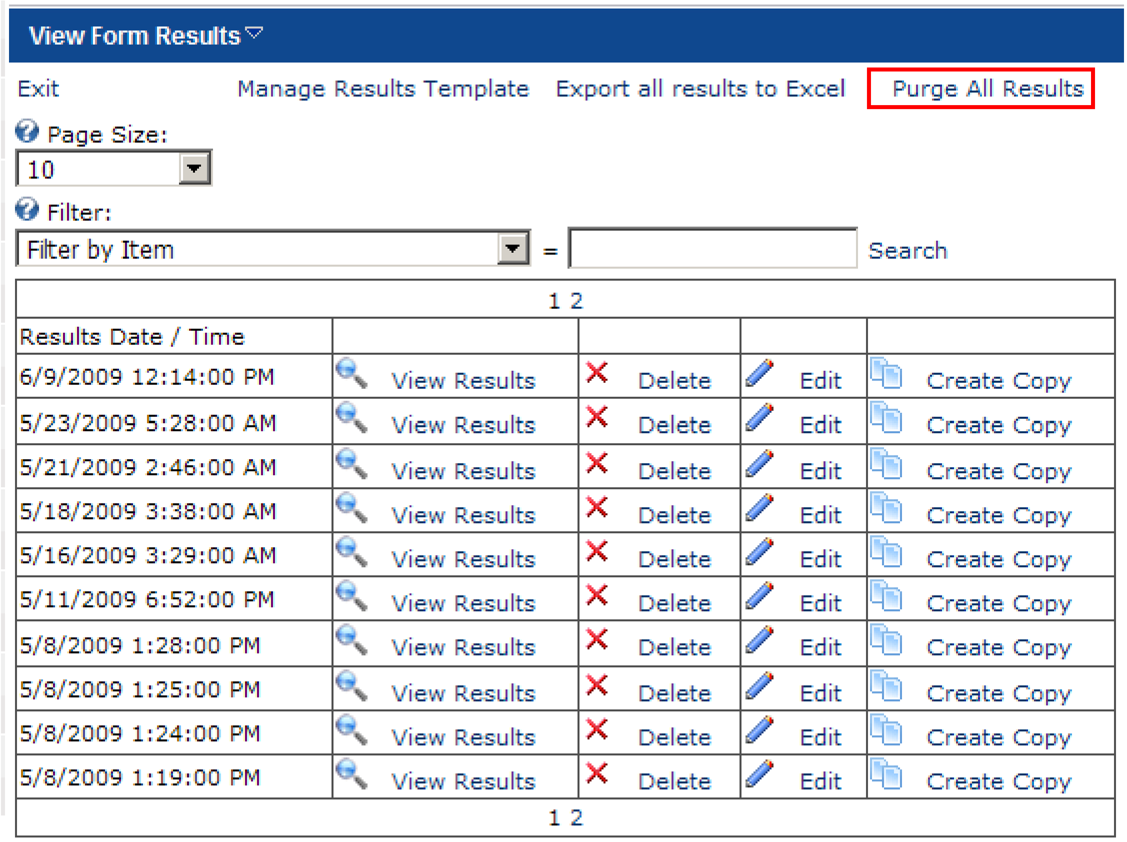


Figure 180: Purging form results

8.2 Managing the Results Template

In order to purge i.e. delete all form results, click the “Purge All Results” link.

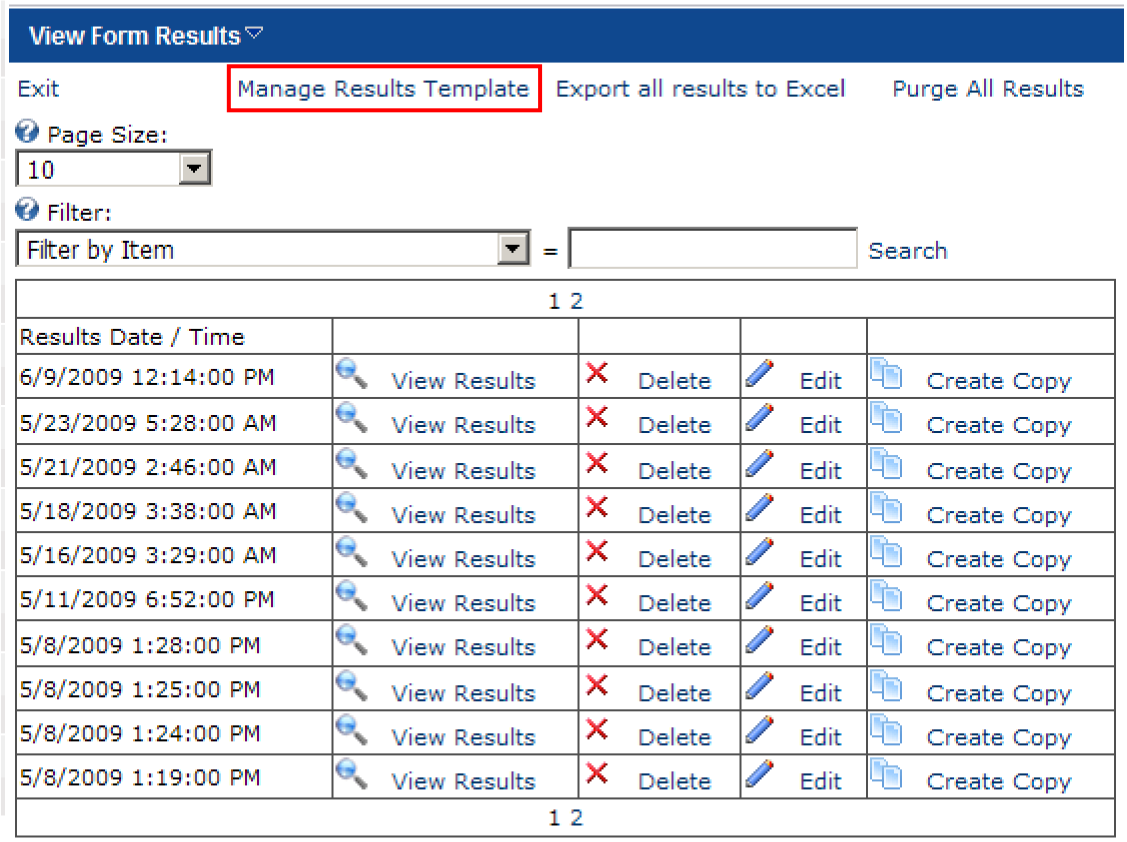


Figure 181: Choosing the "Manage Results Template" option

The following page will be displayed.

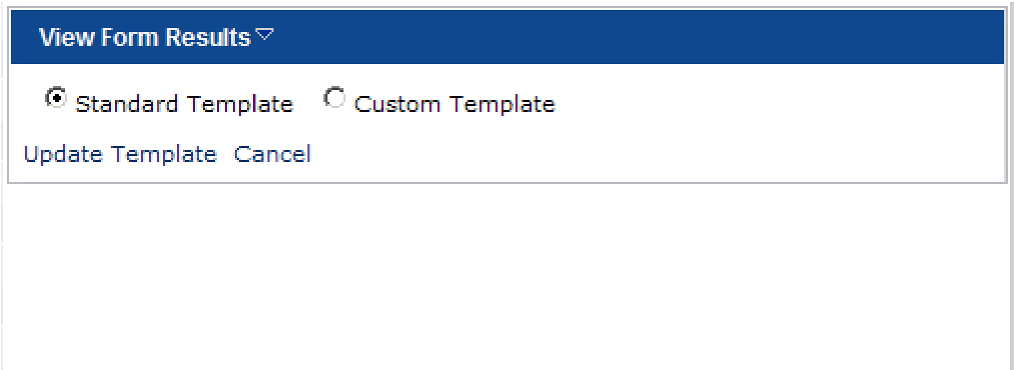


Figure 182: Available form results template options

The following options and parameters are available:

- **Standard Template** – this option is selected by default when the standard template is applied; to modify the template, choose option “Custom Template”
- **Custom Template** – choose this option to begin modifying the form results template according to your preferences; the following screen will be displayed

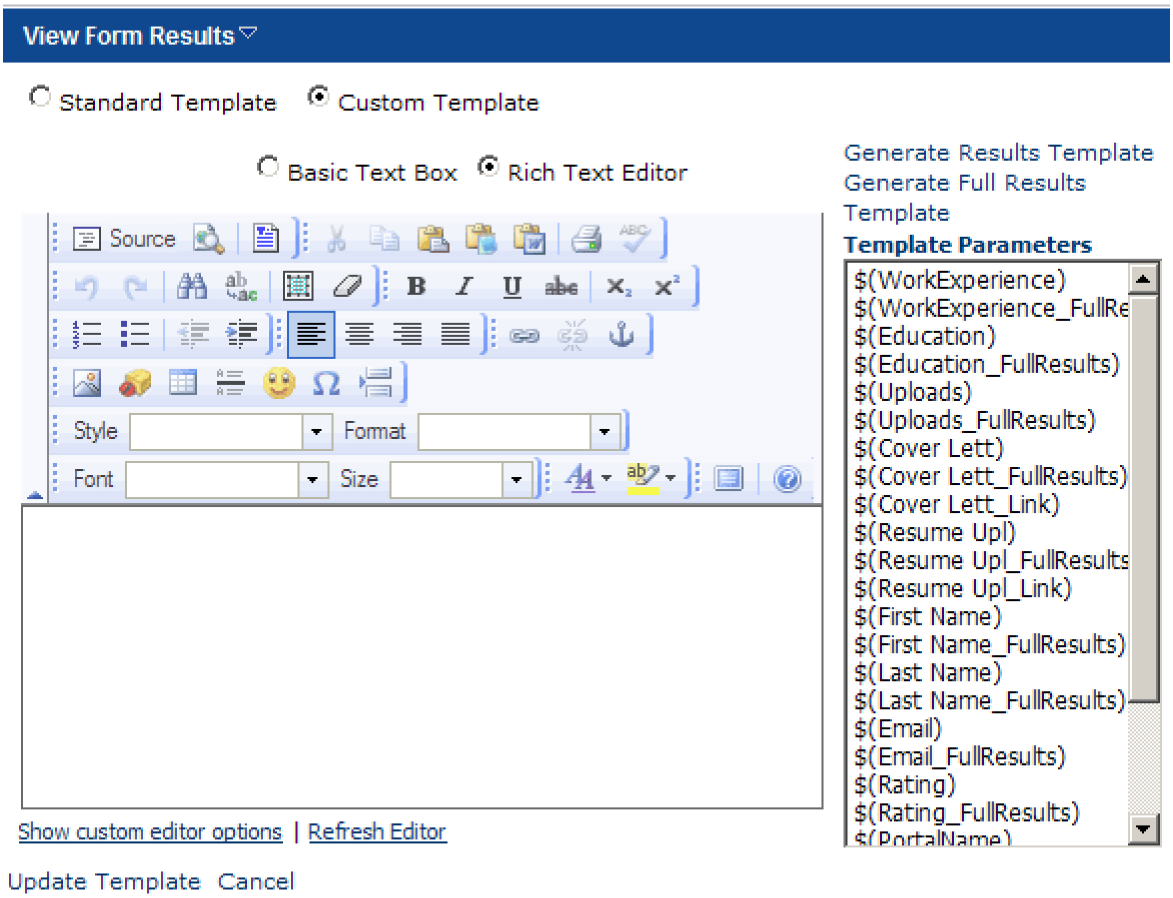


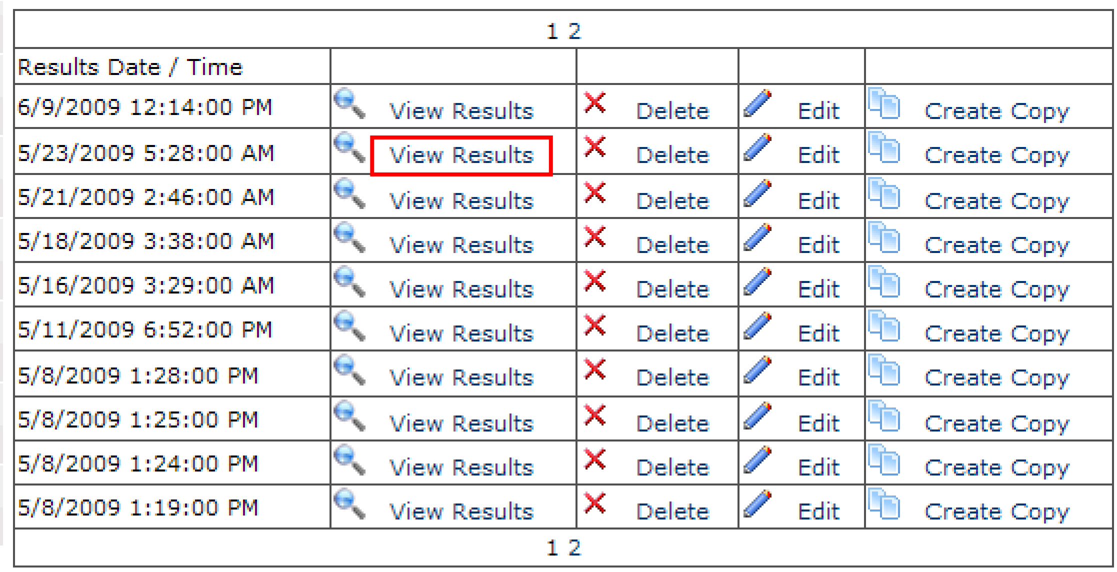
Figure 183: Managing the form results template

The following options and parameters are available:

- **Generate Results Template** – use this option to automatically generate the results template
- **Generate Full Results Template**– use this option to automatically generate the results template
- **Template Parameters** – use the template parameters to create a custom set of parameters which should be displayed within the form

8.3 Viewing a form result

In order to view a form result, click on the “View Results” link next to the desired date.



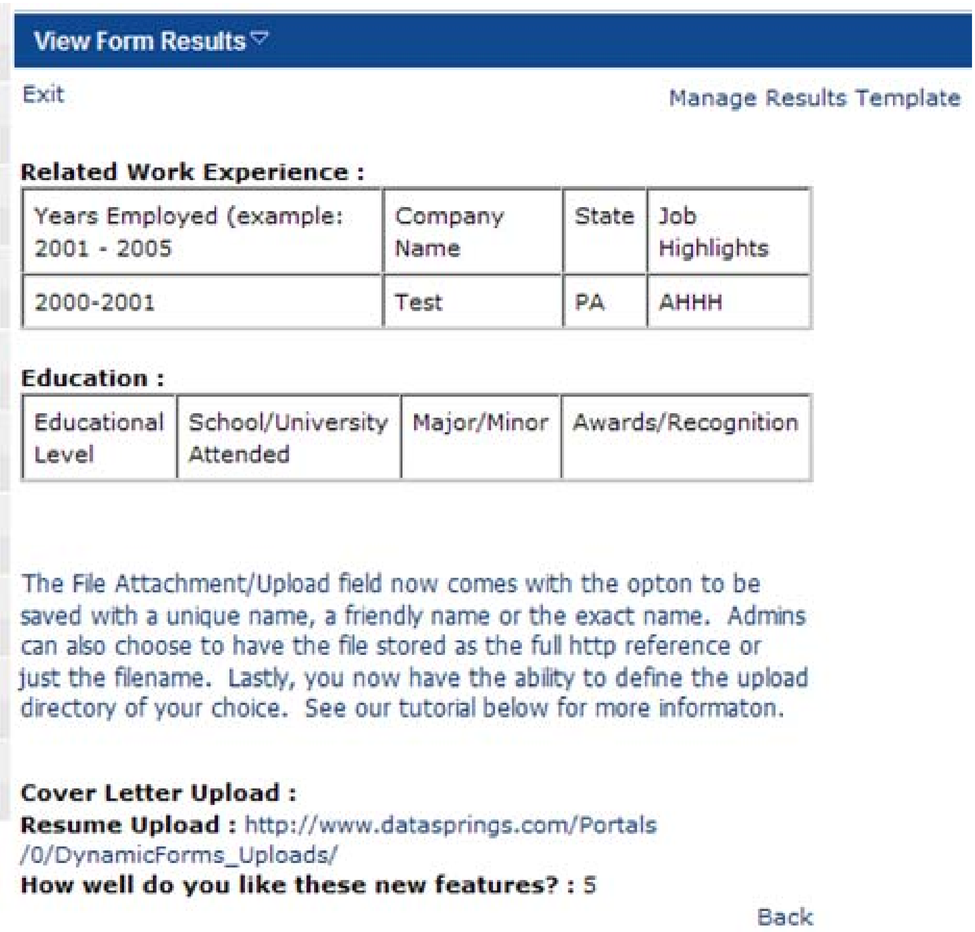
1 2

Results Date / Time				
6/9/2009 12:14:00 PM	View Results	Delete	Edit	Create Copy
5/23/2009 5:28:00 AM	View Results	Delete	Edit	Create Copy
5/21/2009 2:46:00 AM	View Results	Delete	Edit	Create Copy
5/18/2009 3:38:00 AM	View Results	Delete	Edit	Create Copy
5/16/2009 3:29:00 AM	View Results	Delete	Edit	Create Copy
5/11/2009 6:52:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:28:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:25:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:24:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:19:00 PM	View Results	Delete	Edit	Create Copy

1 2

Figure 184: Viewing a form result (step 1/2)

The screen containing detailed form result will be displayed.



View Form Results

Exit Manage Results Template

Related Work Experience :

Years Employed (example: 2001 - 2005	Company Name	State	Job Highlights
2000-2001	Test	PA	AHHH

Education :

Educational Level	School/University Attended	Major/Minor	Awards/Recognition

The File Attachment/Upload field now comes with the opton to be saved with a unique name, a friendly name or the exact name. Admins can also choose to have the file stored as the full http reference or just the filename. Lastly, you now have the ability to define the upload directory of your choice. See our tutorial below for more informaton.

Cover Letter Upload :
Resume Upload : http://www.datasprings.com/Portals/0/DynamicForms_Uploads/
How well do you like these new features? : 5

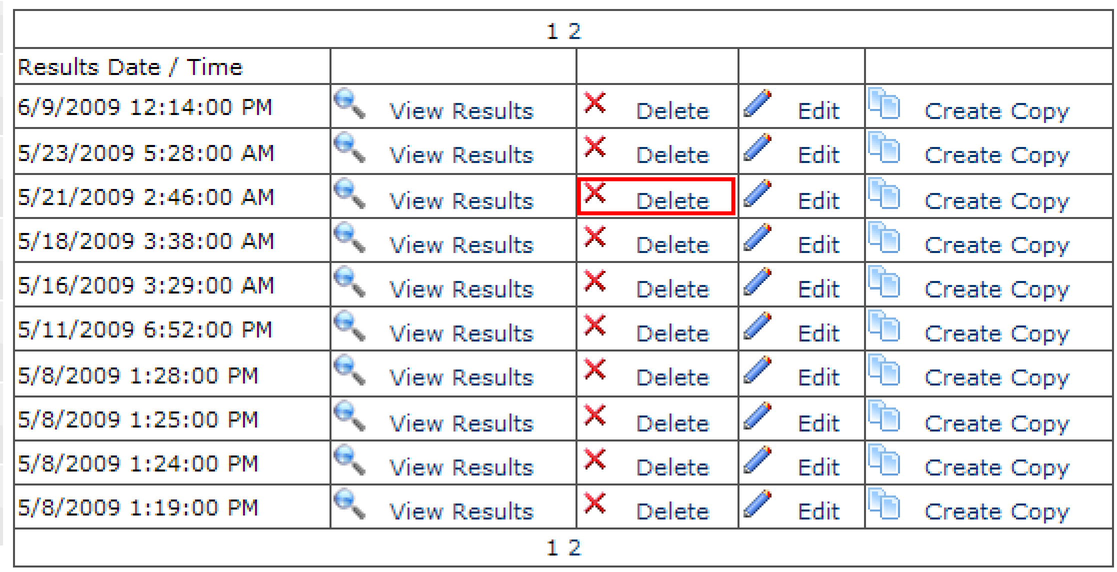
Back

Figure 185: Viewing a form result (step 2/2)

Note: see section 8.7 for further information about exporting results to an Excel file.

8.4 Deleting a form result

In order to view a form result, click on the “Delete” link next to the desired user.



1 2				
Results Date / Time				
6/9/2009 12:14:00 PM	View Results	Delete	Edit	Create Copy
5/23/2009 5:28:00 AM	View Results	Delete	Edit	Create Copy
5/21/2009 2:46:00 AM	View Results	Delete	Edit	Create Copy
5/18/2009 3:38:00 AM	View Results	Delete	Edit	Create Copy
5/16/2009 3:29:00 AM	View Results	Delete	Edit	Create Copy
5/11/2009 6:52:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:28:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:25:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:24:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:19:00 PM	View Results	Delete	Edit	Create Copy
1 2				

Figure 186: Deleting a form result (step 1/2)

Once you click on the “Delete” link the following confirmation dialog will be displayed.

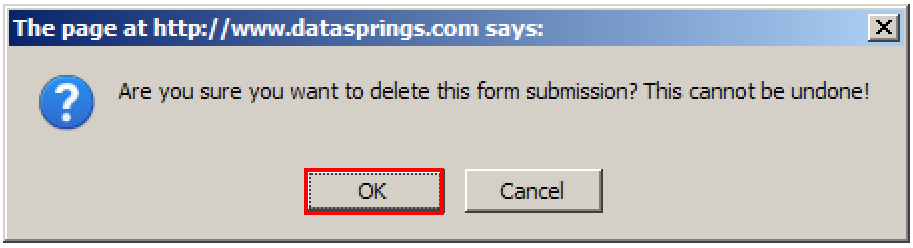
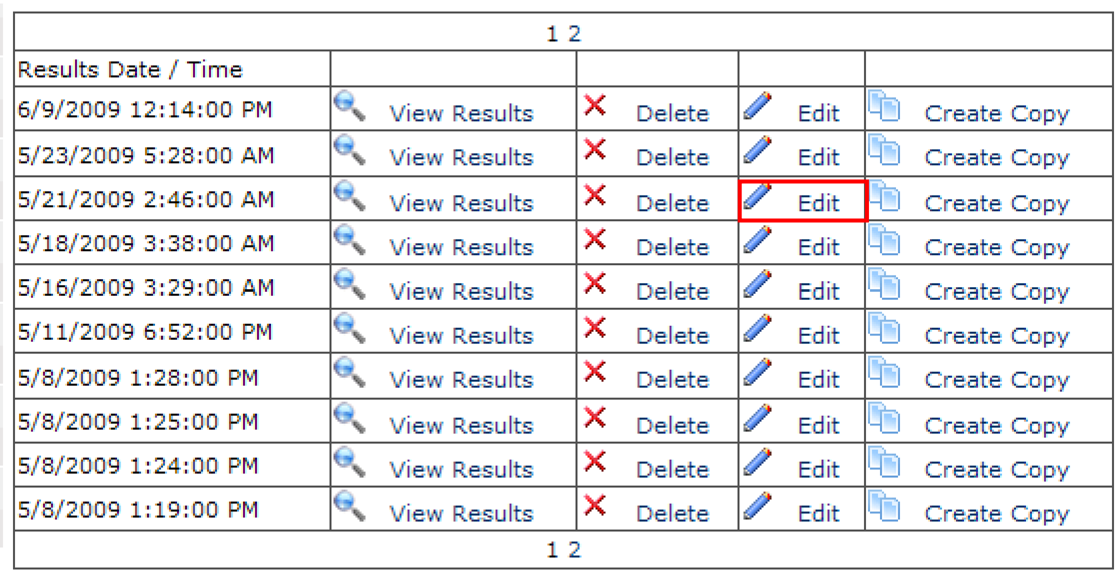


Figure 187: Deleting a form result (step 2/2)

Click “OK” and the selected form submission will be deleted.

8.5 Editing the form result

In order to edit a form result, click the “Edit” link next to the desired result.



1 2				
Results Date / Time				
6/9/2009 12:14:00 PM	View Results	Delete	Edit	Create Copy
5/23/2009 5:28:00 AM	View Results	Delete	Edit	Create Copy
5/21/2009 2:46:00 AM	View Results	Delete	Edit	Create Copy
5/18/2009 3:38:00 AM	View Results	Delete	Edit	Create Copy
5/16/2009 3:29:00 AM	View Results	Delete	Edit	Create Copy
5/11/2009 6:52:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:28:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:25:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:24:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:19:00 PM	View Results	Delete	Edit	Create Copy
1 2				

Figure 188: Deleting a form result (step 1/2)

Once you click on the “Edit” link and the form where you can edit the results will displayed.

8.6 Creating a copy of the result

In order to create a copy of a form result, click the “Create Copy” link next to the desired result.

1 2				
Results Date / Time				
6/9/2009 12:14:00 PM	View Results	Delete	Edit	Create Copy
5/23/2009 5:28:00 AM	View Results	Delete	Edit	Create Copy
5/21/2009 2:46:00 AM	View Results	Delete	Edit	Create Copy
5/18/2009 3:38:00 AM	View Results	Delete	Edit	Create Copy
5/16/2009 3:29:00 AM	View Results	Delete	Edit	Create Copy
5/11/2009 6:52:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:28:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:25:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:24:00 PM	View Results	Delete	Edit	Create Copy
5/8/2009 1:19:00 PM	View Results	Delete	Edit	Create Copy
1 2				

Figure 189: Creating a copy of the result

The copy of the selected form result will be created and displayed.

8.7 Exporting results to Excel

In order to export results to an Excel file, click on the “Export all results to Excel” link.

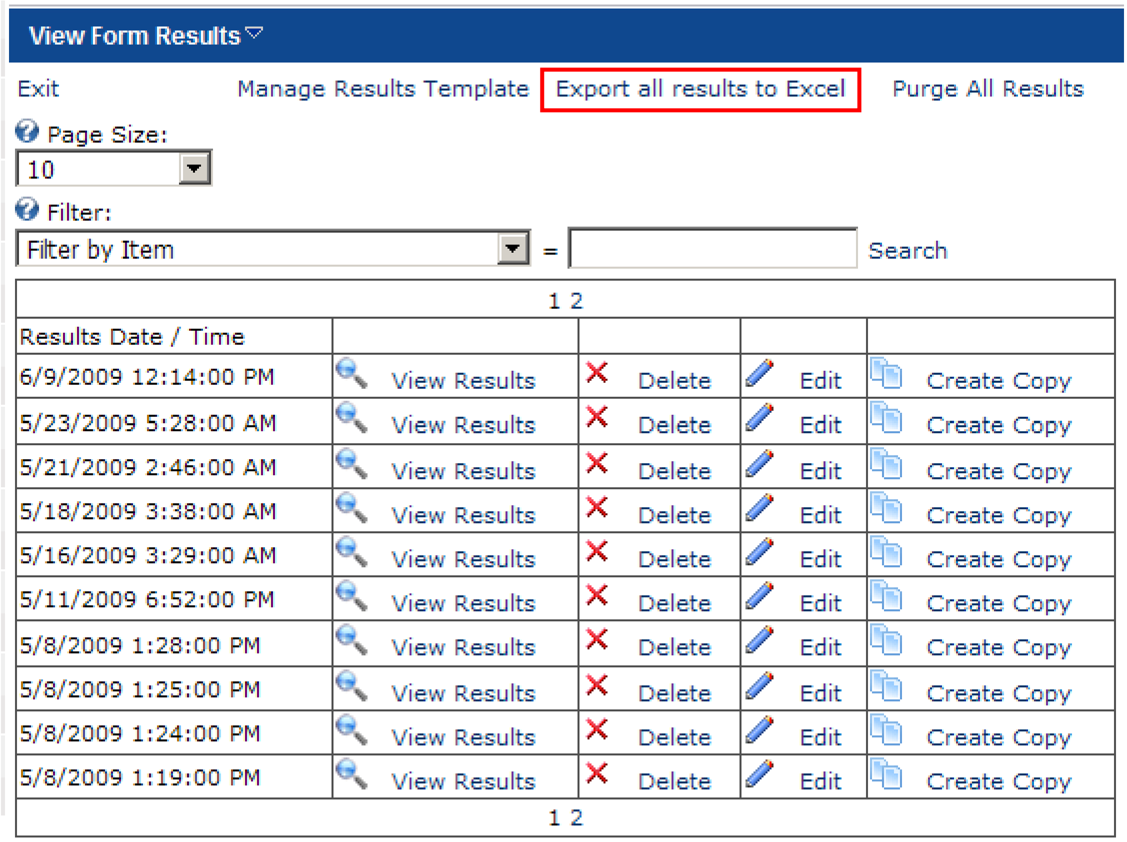


Figure 190: Exporting results to Excel (step 1/3)

The following screen will be displayed.

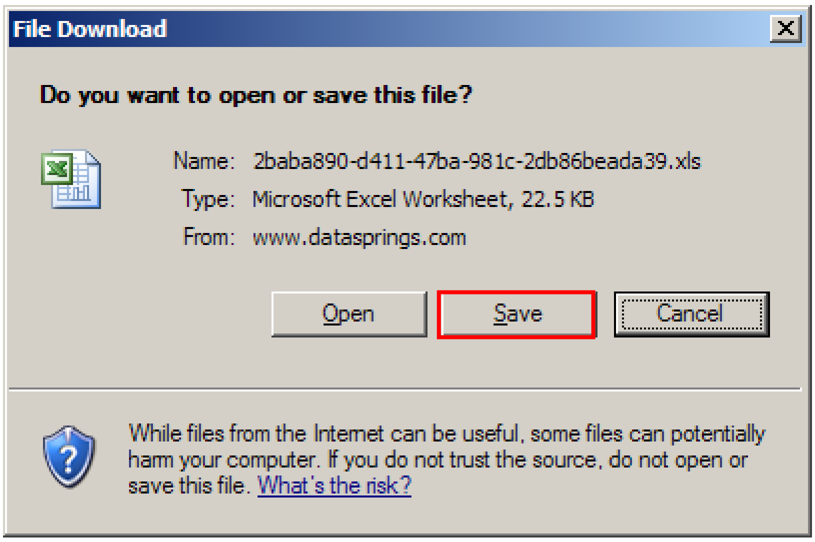


Figure 191: Exporting results to Excel (step 2/3)

Click on the “Save” button in order to save the exported results to your PC and the following screen will be displayed.

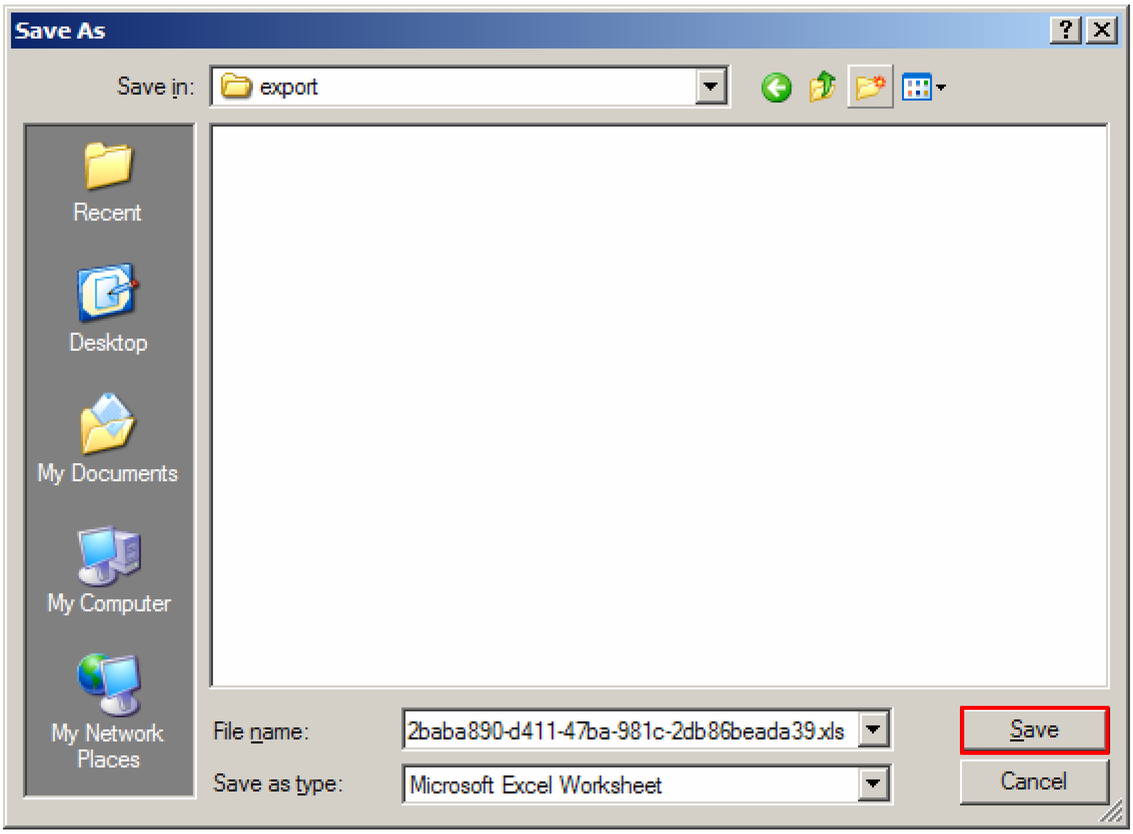


Figure 192: Exporting results to Excel (step 3/3)

Choose the desired location for the exported Excel file and click on the “Save” button in order to save it to your PC. This screenshot demonstrates the layout of the Excel file with exported results.

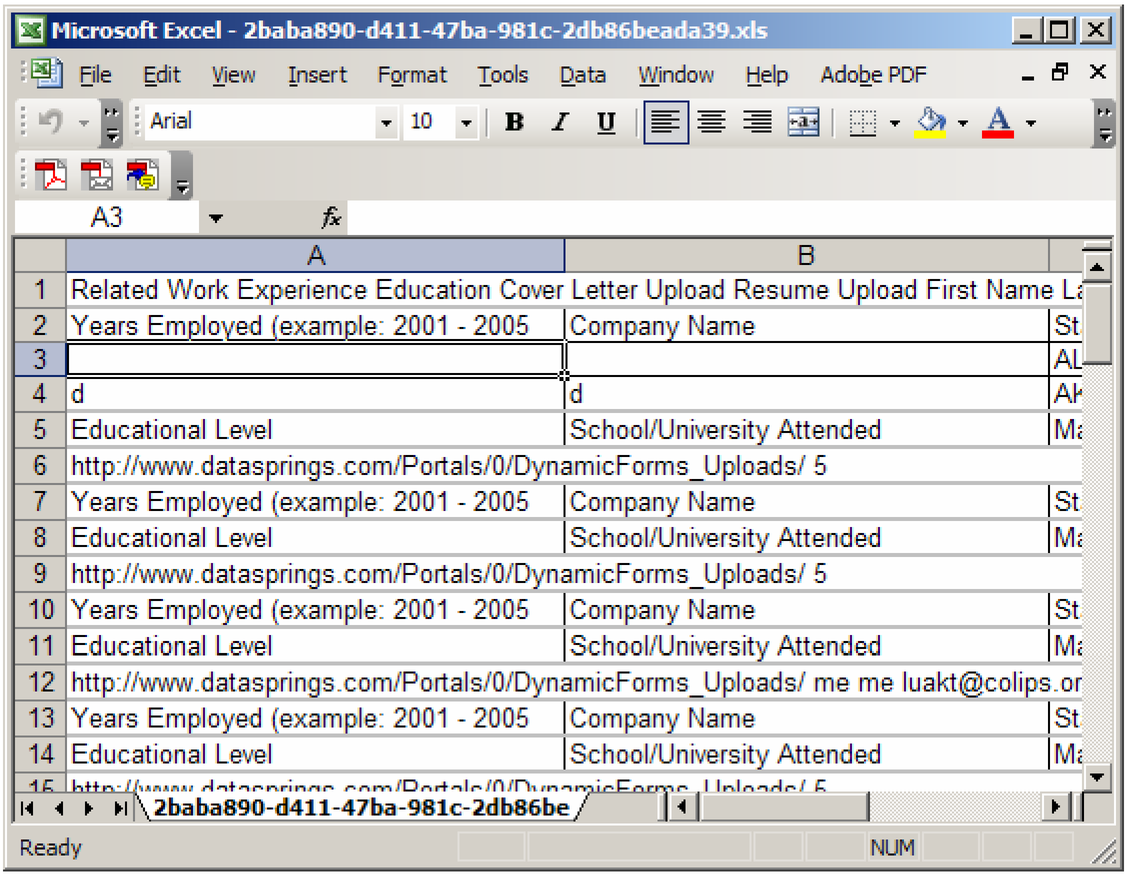


Figure 193: Example of the exported file

9 EXPORTING AND IMPORTING FORMS

The purpose of the export and import options is to allow you to place an already created form on some other page within the website.

The first step is to export the content i.e. export the already created form. Once the content has been exported, you can add the “Dynamic Forms” module to a different page on the website and use the “Import Content” option to add the created form to this page.

9.1 Exporting Content

In order to export the content, choose option “Export Content” from the main menu.

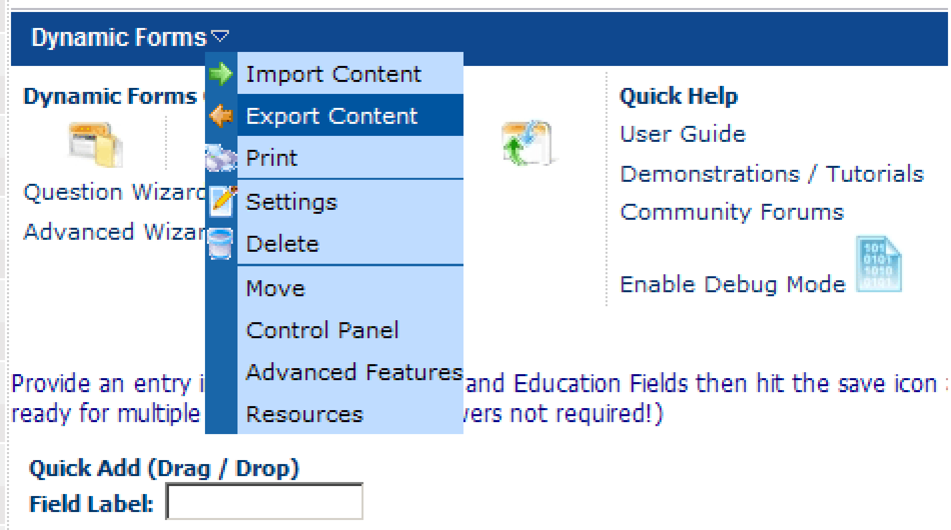


Figure 194: Exporting content (step 1/2)

The following screen will be displayed.

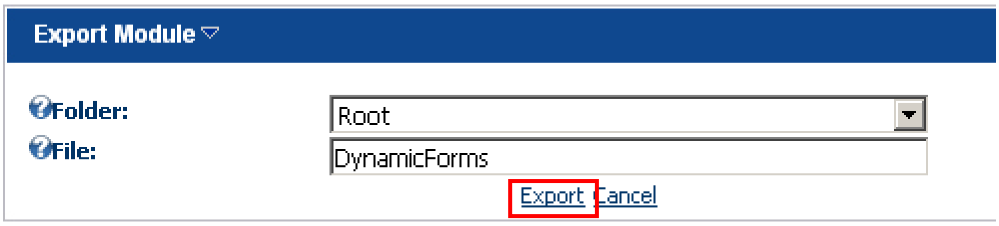


Figure 195: Exporting content (step 2/2)

Choose the desired destination folder and click on the “Export” link. The information about the created form will be exported. The next step is importing the content i.e. form into the desired page (see section 9.2).

9.2 Importing Content

Notes:

- Before you import the content, use the "Export Content" option to export the form first
- After exporting the content, add the "Dynamic Forms" module to a desired page and use the "Import Content" option to add the created form to that page

In order to import the content, i.e. add an already created form to a different page on the website choose option "Import Content" from the main menu.

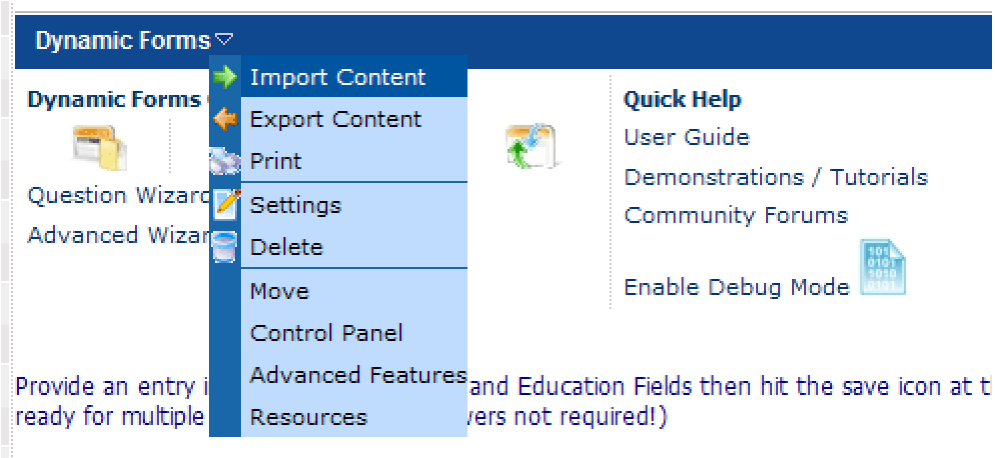


Figure 196: Importing content (step 1/2)

The following screen will be displayed.

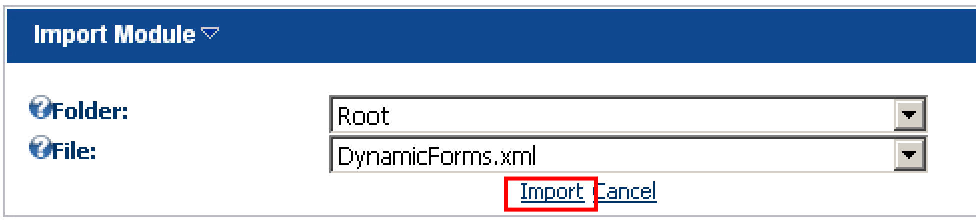


Figure 197: Importing content (step 2/2)

Choose the folder the form has been exported to from the "Folder" pull down menu, then select the desired file and click on the "Import" button. The form will be added to the page.

10 MANAGING SETTINGS

In order to start managing settings choose option “Settings” from the main menu.

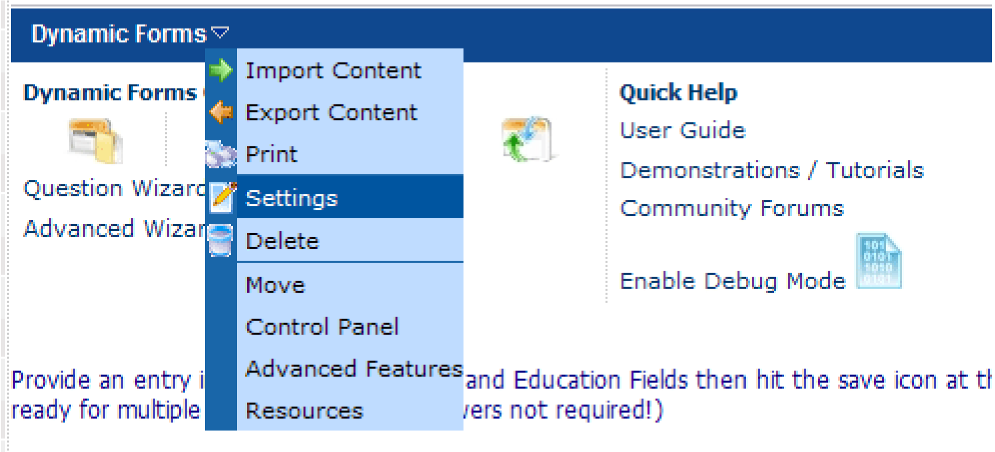


Figure 198: Choosing option "Settings"

The following screen will be displayed.

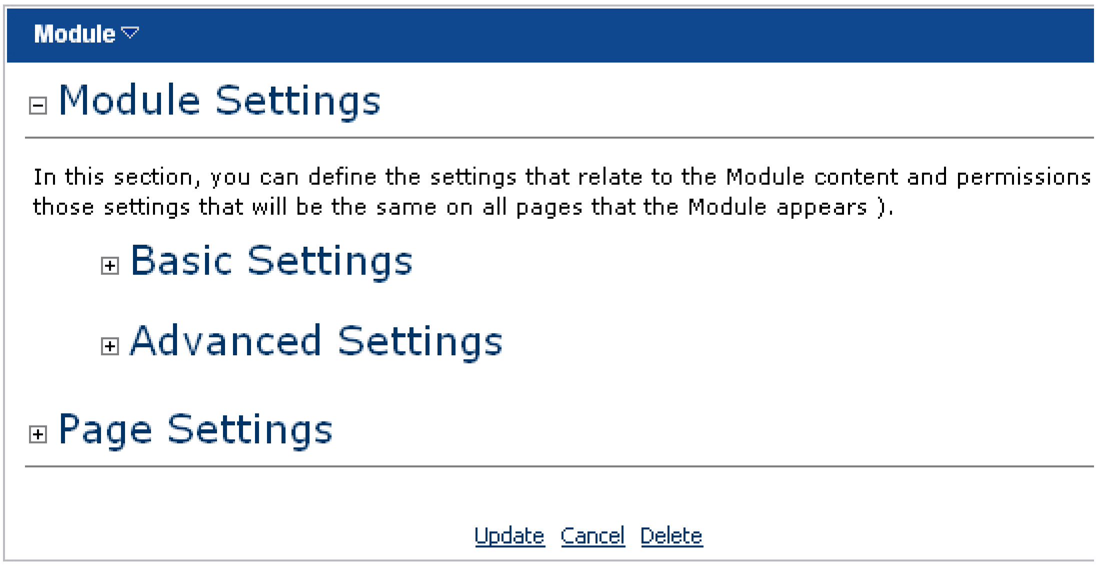


Figure 199: Managing settings

11 DELETING DYNAMIC FORMS MODULE

In order to delete “Dynamic Forms” module, choose option “Delete” from the main menu.

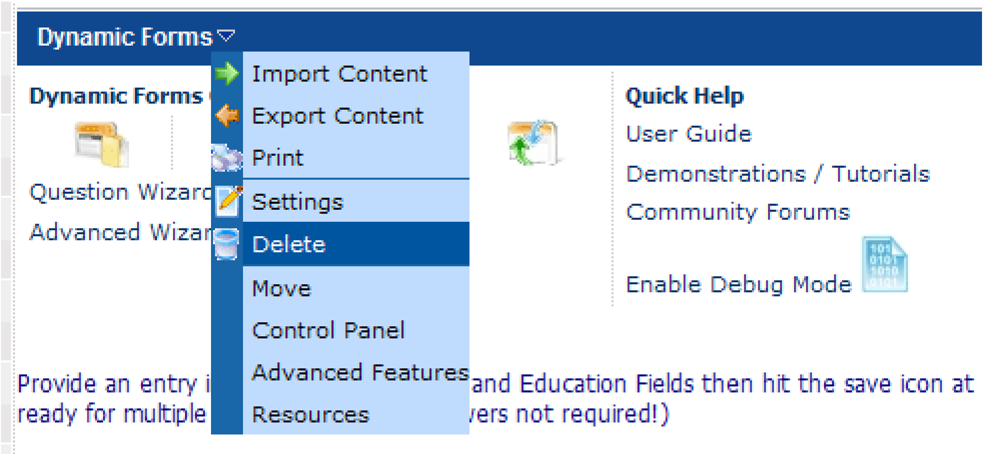


Figure 200: Deleting Dynamic Forms Module (step 1/2)

The following screen will be displayed.

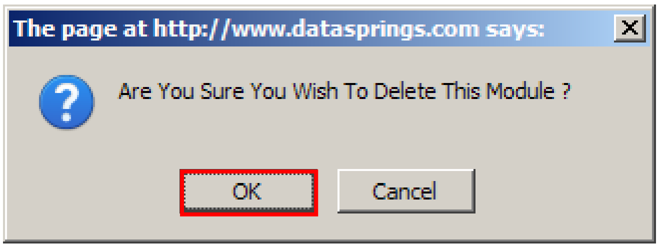


Figure 201: Deleting Dynamic Forms Module (step 2/2)

Click on the “OK” button and the module will be deleted.

12 OTHER RESOURCES

Here is the list of other resources you can use in order to learn as much as possible about different ways you can use the Dynamic Forms module.

12.1 Product Forums:

We encourage you to visit the product forums areas and post support or implementation questions, ideas, enhancement requests, and overall community involvement. Have questions? Often searching the product forums will bring up several threads related to your issue or question.

http://www.datasprings.com/Products/ProductForums/tabid/727/language/en-US/Default.aspx

12.2 Demonstrations:

We encourage you to review the Dynamic Forms demonstrations from the links below for help in getting started. You can download the IPORTABLE XML files, copy these files to your portals home directory and choose 'Import Content' to start from one of these demonstrations:

Dynamic Forms Demonstrations:

http://www.datasprings.com/Products/DNNModules/DynamicForms/DynamicFormsDemonstration1/tabid/754/Default.aspx

Specific Demonstrations:

- **Demonstration #1** - Demonstrates the use of Regular Expressions and Question Events)
- **Demonstration #2** - Demonstrates the use of Regular Expressions and Question Events with Content Localization)
- **Demonstration #3** - Demonstrates the use of Dynamics Forms between multiple user groups)
- **Demonstration #4** - Demonstrates tailored text/HTML integrated with Dynamic Forms to create a payment solution with Paypal)
- **Demonstration #5** - Demonstrates new features including IPortable, Googles 'Auto Fill' toolbar, updated Question Event features)
- **Demonstration #6** - Demonstrates the integration of the Authorizenet Merchant Account)
- **Demonstration #7** - Demonstrates advanced field options and client-side events)
- **Demonstration #8** - Demonstrates form stylesheet properties and layout options)
- **Demonstration #9** - Demonstrates key question and completion events
- **Demonstration #10** - Demonstrates how to create Multi-Page/Wizard functionality using Dynamic Forms
- **Demonstration #11** - Demonstrates how to create data-driven, dependent dropdowns using Dynamic Forms
- **Demonstration #12** - Demonstrates how to perform SQL Validation using Dynamic Forms
- **Demonstration #13** - Demonstrates retrieving data from Dynamic Forms from other 3rd party modules, clear results feature, custom javascript error handling, and inital javascript client side events

- **Demonstration #14** - Dynamic Forms Initial SQL Rendering Demonstration as well as a work flow for create, edit, and delete with integration of a custom table
- **Demonstration #15** - Dynamic Forms PayPal® Integration for Purchase Demonstration
- **Demonstration #16** - Dynamic Forms PayPal® Integration for Subscription Demonstration
- **Demonstration #17** - Dynamic Forms PayPal Integration (optionally turn payment gateway off based on short field name 'EnablePGateway')
- **Demonstration #18** - Dynamic Forms PayPal® Integration (optionally switch between PayPal and Authorizenet payment gateways based on short field name of 'PaymentGateway')
- **Demonstration #19** (The main purpose of this demonstration is to showcase the variety of layout options available with Dynamic Forms. This expanded form showcases the use of column spans, label width, field width, and many others.)
- **Demonstration #20** (Demonstrates a number of calculation methods to obtain totals using various field types such as radio buttons, dropdown menus and checkbox lists.)
- **Demonstration #21** (Similar to demonstration 14, but highlights other 3rd party integration modules for reporting and searching including IndooGrid and Open Web Studio (OWS).)
- **Demonstration #22** (Demonstrates using the new Gridview field type, Rating field type, and new PDF Completion Event as some general tutorials on new features for the 3.0 release)
- **Demonstration #23** (Demonstrates allowing user to save the form for later use, and new 'View' and 'Edit' links as well as some tutorials on new features including the ability to 'Create copy from' results, editing results, and viewing results within a custom template)
- **Demonstration #24** (Demonstrates the new data grid field type and other features such as the ability to choose alternate locations and directories for file upload fields)
- **Demonstration #25** (Demonstrates integration examples of how to integrate Dynamic Forms with other modules such as GeoSprawl Store Locator / Map Module allowing user to save the form data and later display their information on a map)
- **Demonstration #26** (Demonstrates new jQuery features within Dynamic Forms 3.0 such as watermark, masked textbox editor, select all for check box lists, ability to add new items to a combo box dynamically, and other great new jQuery enhancements.)
- **Demonstration #27** (Demonstrates extended work flow functionality and integration with a Tabs module. Within this deomnstratino we highlight how integration works between Dynamic Forms and DNN Aggregator for creating robust forms with easy tabbed navigation.)

12.3 Known Issues

The following are the known issues at the time of 3.0 release at the time of the release (03.00.10):

- Not all Rich Text Editors and DNN® versions support AJAX, therefore if you are using Rich Text Fields or having problems saving emails events or other properties with the Rich Text Editor, you can disable AJAX within module configuration under the Module Configuration, General Settings section.
- There are known issues with AJAX and the file upload/image upload fields. If you are using these fields you should disable AJAX for the form. This is disabled under module configuration, general settings.

- There is a known bug with the Data Grid field type if you also enable the feature to display the label and the field in the same column (instead of left to right). If you have problems with the data grid field please disable the feature to display the label and the field in the same column. Our development team is reviewing this issue.
- Some field types do not currently support the 'Save for later' and 'Edit Results' enhancements. These include the Gridview/DataGrid/File Upload/Image Upload field types.
- Dynamic Forms will generate an 'Error 500' on servers running IIS7 with 64 bit operating systems whenever you click 'Update Settings'. All settings appear to be saved and we are still researching why it generates this error. This should only occur when clicking Update Settings and the settings are saving properly.
- Tokens that include the _Text property such as $(ComboBox_Text) or $(RadioButtonField_Text) do not currently work for PayPal integrations at this time. These variables are generated from the form itself at the time of submission and not saved directly to the database, and since PayPal integration works within a silent 'live' link between PayPal and the form it pulls the value from the database. We are researching ways to handle this in future versions. Additionally, a work around is to use a client side event to save the text property to a hidden field. Please refer to the product forums for examples and assistance to implement this work around.
- If you are signed in as an admin / user with edit rights and have enabled the display after message, you might receive (in edit mode) several JavaScript pop up messages when you submit the form. These messages reference the Drag/Drop ability within Dynamic Forms and we are reviewing a fix (the same JavaScript functions are being called but since the form fields no longer exist its generating a pop up error). This message is only reflected when an admin submits the form in edit mode and uses the display after message, so end users should never see this message.

DATA SPRINGS®

Great Ideas. Always Flowing.™